การวิเคราะห์เนื้อหาการใช้งานอินเทอร์เน็ตของพนักงาน

THE ANALYSIS OF INTERNET CONTENTS ACCESSED BY EMPLOYEES

ทศพล พรมเพชร์

ธนาเดช ภูโพธิ์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2560

การวิเคราะห์เนื้อหาการใช้งานอินเทอร์เน็ตของพนักงาน

THE ANALYSIS OF INTERNET CONTENTS ACCESSED BY EMPLOYEES

ทศพล พรมเพชร์
ธนนภัช สุโพธิ์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2560

ปริญญานิพนธ์ปีการศึกษา 2560

สาขาวิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การวิเคราะห์เนื้อหาการใช้งานอินเทอร์เน็ตของพนักงาน

THE ANALYSIS OF INTERNET CONTENTS ACCESSED BY EMPLOYEES

ผู้จัดทำ

1. นายทศพล พรมเพชร์ รหัสนักศึกษา 57010513

2. นายธนนภัช สุโพธิ์ รหัสนักศึกษา 57010543

รัฐ

อาจารย์ที่ปรึกษา

(อาจารย์รัฐชัย ชาวอุทัย)

การวิเคราะห์เนื้อหาการใช้งานอินเทอร์เน็ตของพนักงาน

นายทศพล พรมเพชร์ 57010513
นายธนนภัช สุโพธิ์ 57010543
ดร.รัฐชัย ชาวอุทัย อาจารย์ที่ปรึกษา
ปีการศึกษา 2560

บทคัดย่อ

ในยุคที่อินเทอร์เน็ตเป็นเครื่องมือในการสืบค้นข้อมูล การเข้าถึงสื่อทุกชนิด การตอบข้อสงสัยต่างๆ เป็นไปได้โดยง่ายและในยุคที่สามารถเข้าอินเทอร์เน็ตได้ทุกที่ และหลายองค์กรเล็งเห็นความสำคัญของข้อมูลของพนักงานในองค์กรที่เป็นปัจจัยที่สำคัญในการขับเคลื่อนองค์กร การได้ทราบถึงองค์ความรู้ที่พนักงานได้สืบค้น เห็นกลุ่มของพนักงานที่สนใจเรื่องเดียวกัน เห็นความสนใจในเรื่องต่างๆ ของพนักงานในแต่ละช่วงเวลา จะช่วยให้ฝ่ายบริหารสามารถวางแผนถึงการจัดการได้ เช่นการจัดอบรมเพื่อพัฒนาพนักงาน ในองค์ความรู้นั้นๆ หรือทราบความคืบหน้าของงาน โดยวิธีการสัมภาษณ์หรือทำแบบสำรวจเป็นวิธีที่ไม่สะดวกสำหรับทุกฝ่ายและสร้างความลำบากใจทั้งผู้ให้ข้อมูลและผู้ทำการสำรวจ เพื่อลดปัญหาเหล่านี้ จึงมีแนวคิดที่จะวิเคราะห์การค้นคว้าของพนักงานจากประวัติการใช้งานอินเทอร์เน็ต เนื่องจากพนักงานมีการใช้อินเทอร์เน็ตอยู่ตลอดเวลาอยู่แล้ว และองค์กรมีสิทธิจากทรัพยากรที่จะมีสิทธิ์เก็บข้อมูลในส่วนนั้น โดยที่พนักงานสามารถเข้าใช้งานตามเงื่อนไขขององค์กรโดยสมัครใจ

ดั้งนั้นทางผู้วิจัยได้ทำปริญญานิพนธ์นี้ขึ้นเพื่อวิจัยข้อมูลที่ซึ่งผู้วิจัยเล็งเห็นว่าการได้นำทักษะด้านการวิเคราะห์ข้อมูลมาช่วยในการนำข้อมูลการใช้งานอินเทอร์เน็ตของบุคลากรจากแผนกใดแผนกหนึ่งในองค์กร แล้วนำมาหาพนักงานหรือกลุ่มของพนักงานที่สนใจหัวข้อสำคัญจากเว็บไซต์ที่พนักงานได้เข้าไปใช้งาน แล้วนำเสนอข้อมูลในรูปแบบของแดชบอร์ด เพื่อดูข้อมูลที่ได้จากการทำการวิเคราะห์จะช่วยเป็นแนวทางสำหรับการวางแผนเพื่อการพัฒนาองค์กรในการทำงานให้ดียิ่งขึ้น

THE ANALYSIS OF INTERNET CONTENTS ACCESSED BY EMPLOYEES

Mr. Thossapol Prompetch 57010513

Mr. Thanonphat Supho 57010543

Dr. Rathachai Chawuthai Advisor

Academic Year 2017

ABSTRACT

Nowadays, the Internet becomes a powerful tool for searching information, accessing to knowledge, and answering any questions. Many departments realize that the organizational knowledge of employees is a key for driving the organization. Knowing the movement of interesting topics of employees and groups of employees who are interested in similar topics in every period of time can help the organization improve the team effectively such as training plan. However, doing interviews and collecting questionnaires for finding interesting topics from each employee are not a good method due to the inconvenient and uncomfortable situations. In order to access these data efficiently, we have an approach to the analysis of the Internet usages of every employee, because they access Internet content every day and the department has right to access the history as well.

For this reason, our study is an attempt to address the according issue. To conduct this research, we use data analysis methods for analyzing the internet usage logging of employees in a department for finding the key terms and interest groups. Then, present the analytical result into the dashboard to be a support system for the management roles to have plans for improving organizational knowledge.

กิตติกรรมประกาศ

สำหรับปริญญานิพนธ์ฉบับนี้สามารถสำเร็จไปได้ด้วยดี ด้วยคำปรึกษาและคำแนะนำจาก ดร. รัฐชัย ชาวอุทัย อาจารย์ที่ปรึกษา ที่ให้ความช่วยเหลือ ให้แนวคิดและความรู้ แนะนำทางเมื่อเกิดปัญหาและชี้แนะจุดบกพร่องของปริญญานิพนธ์ให้สมบูรณ์ยิ่งขึ้น ข้าพเจ้าขอขอบพระคุณเป็นอย่างสูง

ขอขอบพระคุณคณาจารย์ภายในภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ที่ให้ความรู้ต่างๆ ไม่ว่าจะเป็น ซอฟต์แวร์ เน็ตเวิร์ค และฮาร์ดแวร์ ที่เป็นส่วนช่วยในการช่วยทำปริญญานิพนธ์ฉบับนี้ตั้งแต่อดีตจนถึงปัจจุบัน ขอขอบพระคุณเป็นอย่างยิ่ง

ขอขอบคุณเพื่อนๆ ทุกคนที่อาจจะกล่าวถึงหรือไม่ได้กล่าวถึง ที่ให้คำแนะนำ และเป็นที่ปรึกษา และให้ความคิดเห็นของโครงงาน จนกระทั่ง ปริญญานิพนธ์นี้สำเร็จลุล่วงไปได้ด้วยดี

ขอขอบคุณ คุณไพฑูรย์ ชีวินศิริวัฒน์ และคุณณัฐนภัส รชตะวิวรรธน์ ที่ให้คำแนะนำ ทั้งทางด้านประสบการณ์การทำงาน เทคนิคต่างๆ และคำแนะนำทางด้านวิทยาศาสตร์ข้อมูล

ขอขอบคุณห้อง ปฏิบัติการของภาควิชาวิศวกรรมคอมพิวเตอร์ ที่เอื้อเฟื้อสถานที่ โต๊ะ อุปกรณ์

ขอกราบขอบพระคุณบิดา มารดา ครอบครัวที่เป็นที่รักของข้าพเจ้า ที่คอยอบรมสั่งสอน ให้กำลังใจและการสนับสนุน ตลอดจนปริญญานิพนธ์นี้สำเร็จไปด้วยดี และหวังเป็นอย่างยิ่งว่าจะเป็นประโยชน์ต่อผู้ที่นำไปศึกษาต่อหรือเป็นแนวทางในการพัฒนาไปใช้งานจริง

ทศพล พรมเพชร์
ธนนภัช สุโพธิ์

สารบัญ

หน้า

สารบัญ (ต่อ)

หน้า

สารบัญ (ต่อ)

หน้า

สารบัญตาราง

ตาราง **หน้า**

สารบัญรูป

สารบัญรูป (ต่อ)

รูป **หน้า**

สารบัญรูป (ต่อ)

รูป **หน้า**

บทที่ 1
บทนำ

1.1 ความสำคัญและที่มาของโครงงาน

การพัฒนาศักยภาพบุคลากรในองค์กร เป็นปัจจัยที่สำคัญ ที่ทำให้เกิดการขับเคลื่อนองค์กร อาจกล่าวได้ว่าการพัฒนาคน เอื้อประโยชน์ต่อองค์ การพัฒนาบุคลากรจะช่วยเพิ่มผลผลิต และลดต้นทุนให้น้อยลง เนื่องจากบุคลากรมีความเชี่ยวชาญในด้านงานที่ทำมากขึ้น ส่วนของข้อผิดพลาดต่างๆ ก็จะลดลง การพัฒนาบุคลากรจึงเป็นหนทางหนึ่งที่จะช่วยพัฒนาองค์กรให้ก้าวทันต่อความเจริญ โดยประโยชน์โดยตรงต่อตัวบุคลากร เมื่อบุคลากรมีประสิทธิภาพการทำงานที่สูงขึ้น โอกาสความก้าวหน้าในหน้าที่การงานก็จะมากขึ้นด้วย การพัฒนาบุคลากรอย่างต่อเนื่องจะช่วยลดอัตราการลาออกของคนที่มีฝีมือลง เพราะการพัฒนาบุคคลากร ยังจะช่วยให้สร้างความผูกพันระหว่างบุคลากรและองค์กร ก่อเกิดเป็นความทุ่มเท เอาใจใส่ต่อการทำงาน สุดท้ายเพื่อบรรลุเป้าหมายความสำเร็จขององค์กร

เนื่องจากปัจจุบันเว็บไซต์เปรียบเสมือนคลังแห่งความรู้ การสืบค้นการเข้าถึงเว็บไซต์ต่างๆ ช่วยบอกสิ่งที่มนุษย์ต้องการ ในการทำงานของพนักงานในองค์กรส่วนใหญ่ การทราบถึงองค์ความรู้ที่พนักงานกำลังสืบค้น และสามารถวิเคราะห์หาความต้องการของพนักงานได้ ทำให้องค์กรสามารถวางแผน นำผลการวิเคราะห์ที่ได้จากการเตรียมความพร้อม ไม่ว่าจะเป็นการจัดอบรมเพื่อพัฒนา ในองค์ความรู้นั่นๆ หรือทราบความคืบหน้าของงาน ซึ่งข้อมูลดังกล่าวบริษัททางด้านไอทีได้ทำการบันทึกข้อมูลอยู่ตลอดเวลาแต่เนื่องจากปริมาณข้อมูลที่มาก และไม่ได้นำข้อมูลดังกล่าวมาวิเคราะห์อย่างจริงจัง ดังนั้นทางคณะผู้วิจัยมีความสนใจในการนำข้อมูลเพื่อนำไปวิเคราะห์ และนำเสนอผลการวิเคราะห์ที่เหมาะสมต่อการนำไปใช้ในการตัดสินใจ

การทำเหมืองข้อมูลเว็บไซต์ เพื่อสกัดข้อมูลและสารสนเทศ จากเว็บและบริการบนเว็บ เพื่อที่จะได้นำความรู้ที่ได้มาเพื่อนำมาแก้ปัญหาทางตรง หรือทางอ้อม โดย คณะผู้วิจัยวางแผนใช้ เว็บคอนเทนต์ไมน์นิง (Web Content Mining) , เว็บสตรัคเจอร์ไมน์นิง (Web Structure Mining) และเว็บยูสเซสไมน์นิ่ง (Web Usage Mining) เพื่อค้นหารูปแบบโครงสร้างการเชื่อมโยงที่สำคัญและซ่อนอยู่ในเว็บ โดยหาคำสำคัญมาใช้เพื่อจัดกลุ่มเว็บเพจ และใช้สร้างข้อมูลสารสนเทศที่เป็นประโยชน์ การใช้งานอินเทอร์เน็ต และเว็บแอปพลิเคชัน (Web Application) เป็นของคู่กัน ซึ่งผู้ใช้มีความคุ้นเคยและใช้งานได้ง่าย ตอบโจทย์การใช้งาน การใช้เว็บแอปพลิเคชัน เพื่อนำเสนอข้อมูลอย่างง่ายที่สุดด้วยวิธีทาง การแสดงผลข้อมูล (Data Visualization) เพื่อแสดงรายงานให้เข้าใจง่าย จะทำให้น่าสนใจ และมีประโยชน์ต่อการนำไปตัดสินใจ

1.2 วัตถุประสงค์ของโครงงาน

1. เพื่อหาเรื่องที่พนักงานกำลังสนใจอยู่โดยการวิเคราะห์ข้อมูลการใช้งานอินเทอร์เน็ตของพนักงานในองค์กร
2. เพื่อแสดงผลการวิเคราะห์ในรูปแบบที่เหมาะสมต่อการนำไปใช้ในการตัดสินใจจัดกิจกรรมพัฒนาบุคลากร

1.3 ขอบเขตของโครงงาน

1. ข้อมูลการใช้งานอินเทอร์เน็ตพนักงาน 1 แผนก
2. นำข้อมูลมาจากประวัติการใช้งานที่บริษัทสามารถบันทึกได้
3. เนื้อหาของข้อมูลที่อยู่ในหมวดเทคโนโลยี ไม่รวมเนื้อหาจากสื่อสังคมออนไลน์และสื่อบันเทิง

1.4 วิธีการดำเนินการ

1. ทำการสร้างโมเดล (Model) ที่มี ผลลัพธ์เป็น คำสำคัญ (Keyword) และ จัดประเภทของ เว็บเพจ (Webpage) โดยรับ อินพุต (Input) เป็น URL (Uniform Resource Locator)
2. นำเข้าข้อมูลการใช้งาน อินเทอร์เน็ต ของพนักงานในองค์กรในรูปแบบของไฟล์ .csv แล้วนำข้อมูลนั้นไปเป็นอินพุตให้กับโมเดลที่สร้างเพื่อนำเก็บเอาต์พุตของแต่ละเว็บเพจลงในฐานข้อมูล (Database) ของระบบ
3. วิเคราะห์ข้อมูลการใช้งานอินเทอร์เน็ตแยกตามหัวข้อที่พนักงานสนใจเป็นรายคนและในภาพรวม
4. ออกแบบระบบการแสดงผลข้อมูลในรูปแบบของเว็บแอปพลิเคชันลักษณะของแดชบอร์ด (Dashboard) ให้ สอดคล้องกับข้อมูลที่วิเคราะห์ออกมาให้เข้าใจง่าย
5. สร้างเว็บเซิร์ฟเวอร์ (Web server) เพื่อพัฒนาเอพีไอ (API) สำหรับรับส่งข้อมูล จากฐานข้อมูลไปสู่ฝั่ง เว็บแอปพลิเคชัน

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. ผลจากการวิเคราะห์ข้อมูลสามารถนำไปใช้พัฒนาบุคลากรได้จริง
2. ลดการทำแบบสำรวจความต้องการ เต็มใจ หรือไม่เต็มใจ ซึ่งอาจจะได้ข้อมูลที่มีความคาดเคลื่อนต่อความเป็นจริง
3. มีความรู้ความเข้าใจ ทักษะการทำความสะอาดข้อมูล (Clean Data) และ กระบวนการคัดเลือกคุณลักษณะ (Feature Selection)
4. มีความรู้ความเข้าใจ ทักษะ การแสดงผลข้อมูล
5. มีความรู้ความเข้าใจ ทักษะการออกแบบระบบโดยใช้ Python ร่วมกับ Django เพื่อพัฒนาเว็บแอปพลิเคชัน
6. มีความรู้ความเข้าใจ ทักษะการใช้งาน เพื่อสร้างโมเดลวิเคราะห์และเรียนรู้ข้อมูลขนาดใหญ่

1.6 ส่วนประกอบของปริญญานิพนธ์

เนื้อหาของปริญญานิพนธ์ฉบับนี้ประกอบด้วย 4 บท ได้แก่ บทนำ ทฤษฎีที่เกี่ยวข้อง การออกแบบและพัฒนา การทดลองและผลการทดลอง โดยมีรายละเอียดดังนี้

บทที่ 1 บทนำ กล่าวถึง ความสำคัญและที่มาของโครงงาน วัตถุประสงค์ของโครงงาน ขอบเขตของโครงงาน วิธีดำเนินการ และประโยชน์ที่คาดว่าจะได้รับ

บทที่ 2 ทฤษฎีที่เกี่ยวข้อง กล่าวถึงทฤษฎีที่เกี่ยวข้องกับโครงงาน และหลักการทำงาน

บทที่ 3 การออกแบบและพัฒนา

บทที่ 4 ผลการทดลอง

บทที่ 5 บทสรุปและข้อเสนอแนะ ประกอบไปด้วย บทสรุปที่ได้จากการทำปริญญานิพนธ์

1.7 ตารางเวลาของโครงงาน

ตาราง 1. 1 การดำเนินการในภาคเรียนที่ 1

ลำดับ	งานที่ลงมือทำ	สิงหาคม				กันยายน				ตุลาคม				พฤศจิกายน			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1	วางแผนการออกแบบระบบและหาเครื่องมือที่เหมาะสำหรับการช่วยพัฒนาระบบ																
2	ศึกษาการใช้งาน และ ดีแจงโก เพื่อนำไปใช้ประกอบการเรียกใช้ ไลบรารี (Library)																
3	ศึกษาการใช้ JavaScript รวมทั้ง Html และ CSS พื้นฐาน เพื่อนำไปใช้ควบคุมการทำงานของเว็บแอปพลิเคชัน และการควบคุม JavaScript ไลบรารี																
4	ติดต่อบริษัทที่มีข้อมูลเพื่อนำมาใช้วิเคราะห์ข้อมูล																
5	ศึกษาการเขียนโปรแกรมแบบโปรแกรมเชิงวัตถุ (Object Oriented) โดยใช้ภาษา																
6	ศึกษาหลักการเขียนเว็บแอปพลิเคชันสำหรับงาน การแสดงผลข้อมูล																
7	ศึกษาการใช้ ไลบรารี ตัดคำของ เพื่อใช้ในการตัดคำที่อยู่ในเว็บเพจ																
8	ศึกษาการหารากศัพท์คำเพื่อลดจำนวนของคำที่มีลักษณะซ้ำกัน จนกระทั่งได้คำสำคัญจากเว็บเพจทั้งหมด																
9	ออกแบบส่วนต่อประสานกับผู้ใช้ (User Interface) ของเว็บสำหรับการนำเสนอข้อมูล																

ตาราง 1. 2 การดำเนินการในภาคเรียนที่ 1 (ต่อ)

ลำดับ	งานที่ลงมือทำ	สิงหาคม				กันยายน				ตุลาคม				พฤศจิกายน			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
10	ออกแบบ ตัวส่งข้อมูลหลังบ้านโดยใช้ Python Framework																

ตาราง 1.3 การดำเนินการในภาคเรียนที่ 2

ลำดับ	งานที่ลงมือทำ	มกราคม				กุมภาพันธ์				มีนาคม				เมษายน			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1	วางแผนการออกแบบระบบและหาเครื่องมือเพื่อหาแนวทางใหม่สำหรับการสกัดคำสำคัญที่เป็นภาษาไทย																
2	นำข้อมูลจริงที่ได้จากบริษัทมาทำการเข้าสู่ระบบ																
3	ทำการสร้างโมเดลในการตัดคำและหาคำสำคัญ ทั้งภาษาไทยและอังกฤษ																
4	เขียน เว็บแดชบอร์ด (Web Dashboard) สำหรับ แสดงผลข้อมูลของคำสำคัญ																
5	ทำ เว็บเอพีไอเซิร์ฟเวอร์ (Web API Server) สำหรับ เอพีไอเซอร์วิส (API Service) การค้นคำสำคัญและการแสดงผลอื่น ๆ ให้กับหน้าเว็บแดชบอร์ด																
6	ทดสอบการทำงานระหว่าง เว็บเอพีไอเซิร์ฟเวอร์ กับหน้าเว็บแดชบอร์ด																
7	ปรับปรุงการทำงานระหว่าง เว็บเอพีไอเซิร์ฟเวอร์ กับหน้าเว็บแดชบอร์ด เพื่อเพิ่มประสิทธิภาพการแสดงผล																

บทที่ 2
ทฤษฎีที่เกี่ยวข้อง

2.1 การตัดคำภาษาอังกฤษ (Word Tokenize)

การตัดคำในภาษาไทยตัดโดยใช้อัลกอริทึม คอนโวลูชันแนลนิวลอเน็ตเวิร์ค (Convolutional Neural Network) ทำนายผลด้วยวิธี ไบนารี คลาสสิฟิเคชัน (binary classification) คือทำนายว่าตัวอักษรเป็นตัวเริ่มต้นของคำหรือไม่เช่น

"ตัดคำได้ดีมาก"

สามารถตัดคำได้เป็น "ตัดคำ", "ได้", "ดี", "มาก"

2.2 การตัดคำภาษาอังกฤษ (Word Tokenize)

การตัดคำในภาษาอังกฤษจากประโยค ตัดโดยใช้การเว้นวรรคในประโยค และจบประโยค ด้วยจุด ซึ่งจะสามารถแบ่งแยกคำแต่ละคำได้เช่น

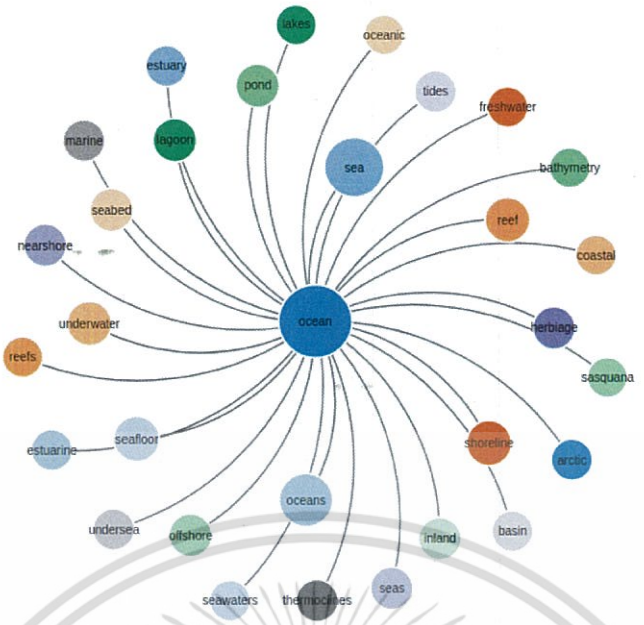
"This is a test."

สามารถตัดคำได้เป็น "This" , "is" , "a" , "test" , "."

2.3 การหารากศัพท์ของคำ ภาษาอังกฤษ

ในการตัดคำ คำที่มีรากศัพท์เดียวกันให้นับเป็น 1 คำ เช่นคำว่า Protection, Protected, Protector, Protecting, Protects ล้วนมีรากศัพท์เป็นคำว่า Protect ดังนั้นในการตัดคำ ให้ตัดคำ ลงท้ายให้เหลือแต่รากศัพท์ของคำนั้น วิธีการนี้เรียกว่า เวิร์ดสเต็มมิ่ง (word stemming) แต่อัลกอริทึมนี้มีจุด ผิดพลาดในการตัดคำเช่นคำว่า leaves จะตัดได้เป็น leav ซึ่งรากศัพท์จริงๆ ของคำนี้คือ leaf

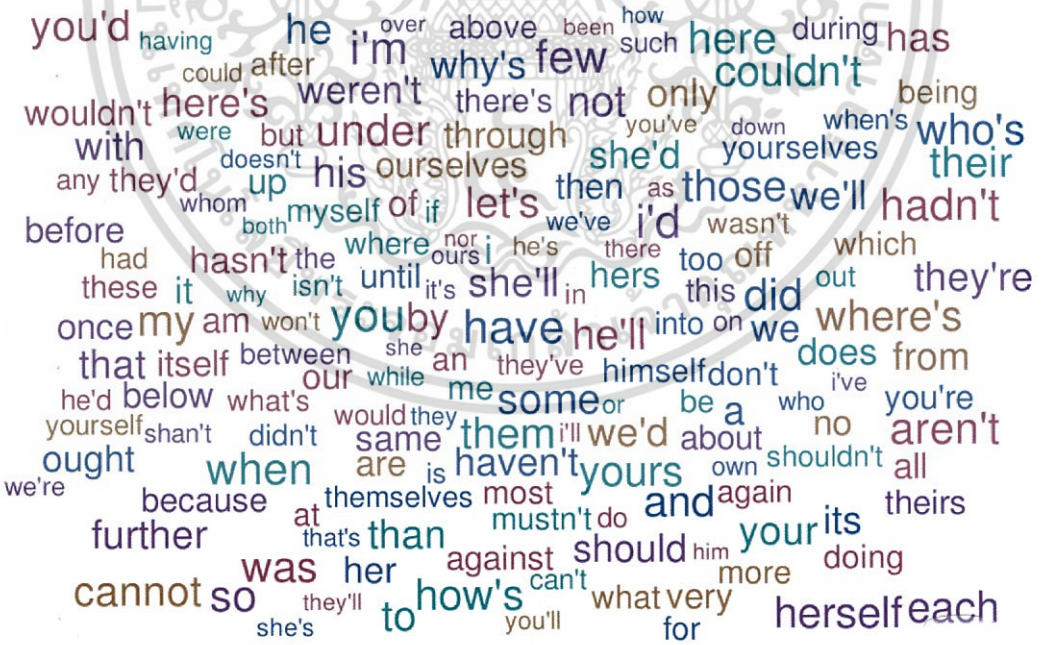
อีกหนึ่งวิธีในการหารากศัพท์ของคำคือ เวิร์ดเล็มมาทิเซชัน (word lemmatization) ให้ผลได้ดีกว่าวิธีแรกคือระวังเรื่องความหมายของคำ โดยมีฐานข้อมูลของคำศัพท์ (lexical database) ที่เรียกว่า เวิร์ดเน็ต (wordnet) มีคำที่มีความหมายคล้ายกัน (synonyms) ซึ่งจะสามารถให้รากศัพท์ที่แม่นยำได้ เช่นคำว่า Ocean กับ sea ก็นับเป็นศัพท์ที่มีความหมายคล้ายกัน



รูป 2.1 ตัวอย่างคำที่มีความหมายคล้ายกับคำว่า ocean

2.4. คำหยุด (Stopwords)

หรือคำที่ปรากฏเป็นจำนวนมากและไม่ให้ความหมายกับประโยค ซึ่งในการทำการประมวลผลภาษาธรรมชาติจะทำการคัดคำหยุดออกไป



รูป 2.2 ตัวอย่างคำหยุดในภาษาอังกฤษ

รูป 2.3 ตัวอย่างคำหยุดในภาษาไทย

2.5 การหาคำสำคัญจากเอกสาร (Keyword Extraction)

2.5.1 TF-IDF

TF-IDF ย่อมาจากคำเต็มว่า term frequency-inverse document frequency โดยที่ TF ย่อมาจาก term frequency คือ จำนวนของความถี่ที่นับได้แต่ละเทอม (term) จาก 1 เอกสาร และ IDF ย่อมาจาก inverse document frequency คือ ส่วนกลับของความถี่ที่นับได้แต่ละเอกสาร เมื่อรวมกัน TF-IDF คือ ตัวเลขทางสถิติที่บ่งบอกถึงความสำคัญของคำสำคัญที่อยู่ในกลุ่มเอกสารหรือคลังข้อมูลคำศัพท์ (Corpus) โดยคำที่มีการปรากฏขึ้นน้อย อาจเป็นคำสำคัญของเอกสารนั้น เมื่อเทียบกับ คำที่ปรากฏขึ้นมาทุกกลุ่มเอกสารจำนวนมากๆ ซึ่งคำเหล่านั้นเรียกว่าคำหยุด (stop word) เช่น a, and, the, or, for, is, are เป็นต้น ดังนั้นการให้น้ำหนักคำ ๆ หนึ่งในเอกสารจะคิดจากสูตร

$$\text{TF-IDF} = \text{TF} \times \text{IDF} \quad (2.1)$$

$$\text{TF} = \text{จำนวน keyword นั้นในเอกสาร} / \text{จำนวน keyword ทั้งหมดในเอกสาร} \quad (2.2)$$

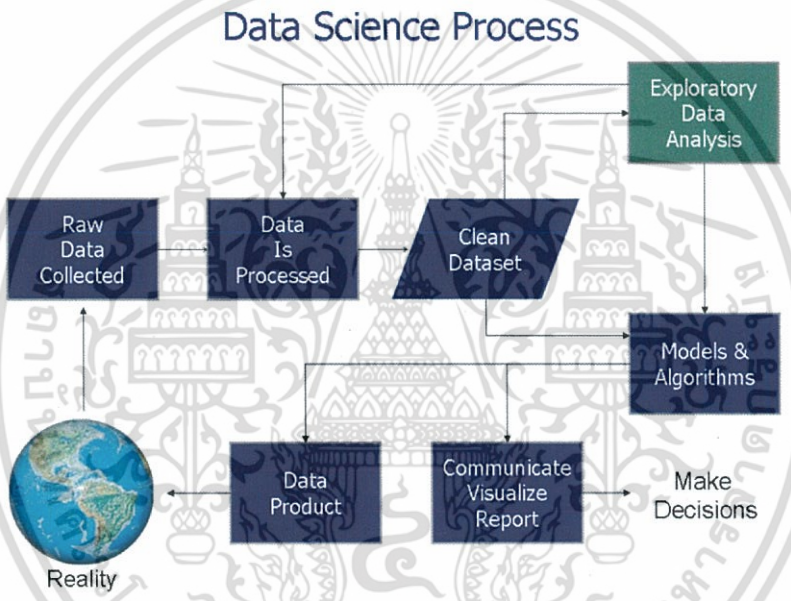
$$\text{IDF} = \log(\text{จำนวนเอกสารทั้งหมด} / \text{จำนวนเอกสารที่ keyword นั้นปรากฏ}) \quad (2.3)$$

ตัวอย่างในเอกสารหนึ่งมี คำสำคัญ cat และ dog ซึ่งไม่อยู่ในลิสต์ของ คำหยุด และมีจำนวนเอกสาร ในระบบทั้งหมด 100 เอกสาร คำสำคัญ cat ปรากฏในเอกสาร 10 ครั้ง และปรากฏในเอกสารอื่น 24 เอกสาร คำสำคัญ dog ปรากฏในเอกสาร 30 ครั้ง และปรากฏในเอกสารอื่น 69 เอกสาร TF-IDF ของ cat = (10 / 100) x log(100 / (24 + 1)) = 0.0602 และ TF-IDF ของ dog = (30 / 100) x log(100 / (69 + 1)) = 0.0465

สรุปได้ว่า คำสำคัญ ที่มีค่าน้ำหนักมากที่สุดคือ cat แม้คำว่า cat จะปรากฏในเอกสารนี้น้อย ครั้งแต่ก็ปรากฏในเอกสารอื่นน้อยครั้งด้วย ทำให้มีค่าน้ำหนักมากกว่า คำสำคัญ dog จึงสามารถใช้ cat เป็นคำสำคัญของเอกสารนี้ได้

2.6 กระบวนการทำวิทยาศาสตร์ข้อมูล (Data Science)

กระบวนการทำวิทยาศาสตร์ข้อมูล คือศาสตร์ที่ว่าด้วยการดึงความรู้หรือข้อมูลเชิงลึก (insight) เพื่อวิเคราะห์ เหตุการณ์จริงจากข้อมูล มีการใช้เทคนิคจากสาขาวิชาคณิตศาสตร์ สถิติ วิทยาการคอมพิวเตอร์ โดยเฉพาะในหัวข้อที่เกี่ยวกับ แมชชีนเลิร์นนิ่ง (machine learning), การทำเหมืองข้อมูล (data mining) และ การแสดงผลข้อมูล (data visualization)



รูป 2.4 กระบวนการทำงานการวิเคราะห์ข้อมูล

2.7 การเก็บรวบรวมและทำความสะอาดข้อมูล (Data Collection and Clean Data)

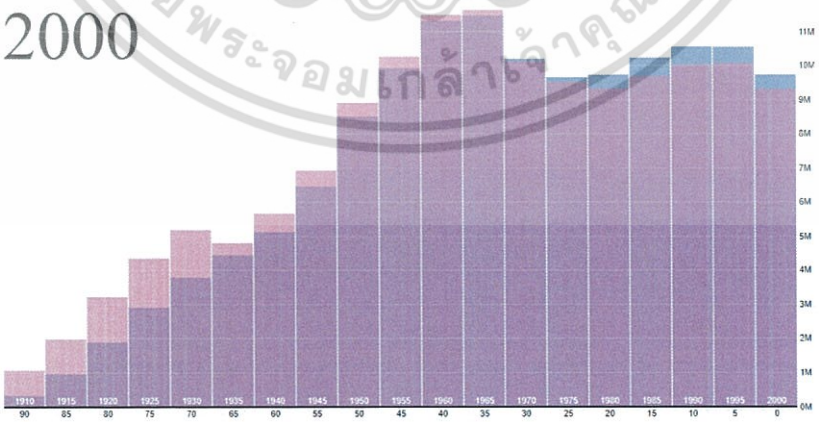
คือการเก็บข้อมูลจากเหตุการณ์จริงเช่น ข้อมูลการใช้ อินเทอร์เน็ต ของพนักงานภายในองค์กร ข้อมูล ธุรกรรม ของธนาคาร เป็นต้น เพื่อที่จะนำข้อมูลไปวิเคราะห์ต่อไป และการทำความสะอาดข้อมูล (Clean Data) คือการทำข้อมูลดิบให้เป็นข้อมูลที่พร้อมนำไปใช้ เพราะข้อมูลดิบจะมีส่วนที่ไม่ต้องการอยู่มาก จึงต้องทำการคัดส่วนนั้นออกไป

2.8 การวิเคราะห์ข้อมูล (Data Analysis)

หลังจากผ่านกระบวนการ การเก็บรวบรวมและทำความสะอาดข้อมูล เรียบร้อยแล้ว นำข้อมูลชุดนั้น มาวิเคราะห์ด้วยวิธีการทางสถิติมาอธิบายพฤติกรรมของข้อมูล รวมถึงการทำ แบบจำลองข้อมูล (Data Model) เพื่อใช้ประโยชน์จากข้อมูลโดยใช้ อัลกอริทึม (Algorithms) ต่างๆของ การเรียนรู้ของเครื่อง เช่น การนำข้อมูลไปทำนายอนาคต (Prediction) ระบบแนะนำ (Recommended) โดยตัวอย่างของโมเดล ที่เป็นที่นิยม เช่น นาอีฟเบย์ (Naive Bayes) , แรนดอมฟอเรส (Random Forest) , และซัพพอร์ตเวกเตอร์แมชชีน (Support Vector machines) เป็นต้น

2.9 การแสดงผลข้อมูล (Data Visualization)

จากคำว่า ว่า Data หมายถึง ข้อมูลหรือสารสนเทศ และ Visualization หมายถึง การถูกทำให้มองเห็น หรือการแสดงภาพ จะได้ว่า คำว่า Data Visualization ซึ่งหมายถึง เทคนิคที่สามารถนำข้อมูลหรือสารสนเทศใดๆ ขึ้นมาสื่อสาร โดยให้อยู่ในรูปแบบที่สามารถมองเห็นได้ โดยทั่วไปจะอยู่ในรูปกราฟ ต่างๆ เช่น กราฟแท่ง กราฟวงกลม กราฟเส้น กราฟจุด ฯลฯ หรืออาจไม่ใช่กราฟโดยตรง ตัวอย่างเช่น แผนที่ โนดเครือข่าย กลุ่มตัวหนังสือ ฯลฯ โดยเป้าหมายหลักของ การแสดงผลข้อมูล คือ การสื่อสารอย่างง่ายที่สุดที่มนุษย์จะสามารถเข้าใจถึงสิ่งกำลังจะสื่อได้ในทันทีที่มองเห็น โดยในปัจจุบันมีเครื่องมือสำหรับการทำ การแสดงผลข้อมูลมากมาย และถูกพูดถึงแพร่หลายในการนำมาแสดงผลจากการวิเคราะห์ข้อมูล เนื่องจากข้อมูลต่างๆ ที่ได้จากการวิเคราะข้อมูลมักเป็นตัวเลขในเชิงปริมาณที่ไม่สามารถเข้าใจได้สำหรับคนที่ไม่เกี่ยวข้องกับการวิเคราะห์ข้อมูล ดังนั้นการนำ การแสดงผลข้อมูลมาประกอบการนำเสนอทำให้เป็นจุดที่น่าสนใจมากขึ้น โดยมีตัวอยากการทำ การแสดงผลข้อมูล ดังนี้



รูป 2.5 การกระจายตัวจำนวนคนและอายุของคนในสหรัฐอเมริกาในช่วงปีต่างๆ

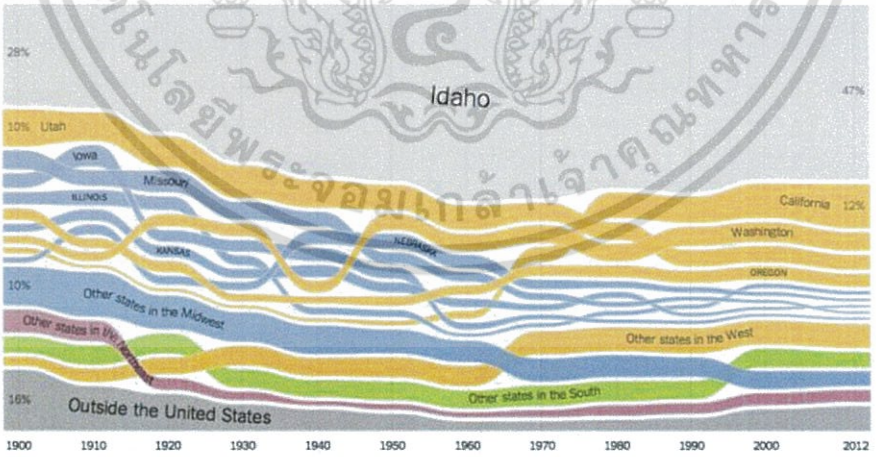
(ที่มา : https://strongriley.github.io/d3/ex/population.html)

จากรูป 2.5 ตัวอย่างนี้เป็น การนำเสนอการกระจายตัวจำนวนคนและอายุของคนในสหรัฐอเมริกาในช่วงปีต่างๆ ระหว่างปี 1910 ถึง ปี 2000 สีชมพูคือ ผู้หญิง และ สีฟ้าคือผู้ชาย โดยใช้ กราฟฮีสโตแกรม (Histogram chart)

รูป 2.6 คำที่ถูกพบมากที่สุดโดยใช้การวางของคำจนเกิดเป็นรูป

(ที่มา : https://www.jasondavies.com/wordcloud/)

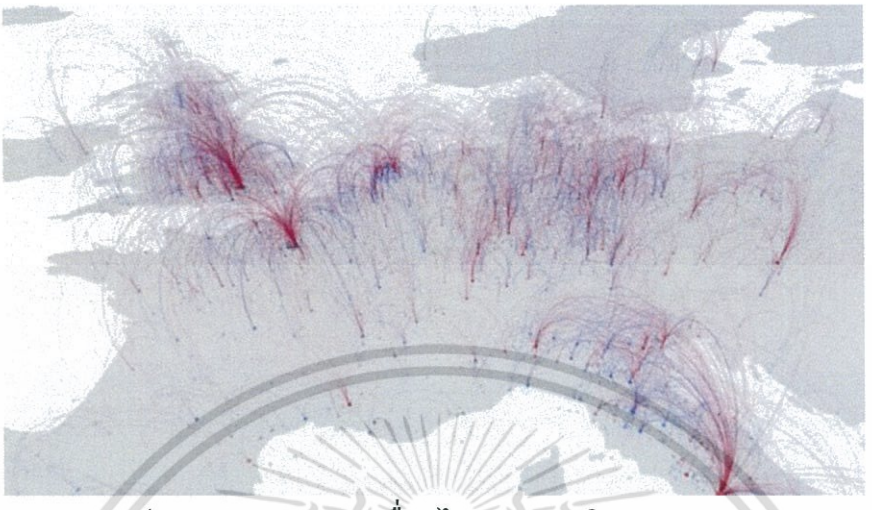
ตัวอย่างต่อมาเป็นการบอกคำที่ถูกพบมากที่สุดโดยใช้การวางของคำจนเกิดเป็นรูป จากรูปที่ 2.6 ตัวอย่างคำที่เด่นชัดที่สุด คือ word , words , sprite ,layout โดยคำที่ขนาดฟอนต์เล็กๆ ก็จะเป็นคำที่มีความถี่น้อยในหน้าเอกสารนั้นๆ เป็นต้น



รูป 2.7 ภาพแสดงข้อมูลการอพยพย้ายถิ่นของประชาชน โดยแสดงอยู่ในรูปแบบของ แผนภูมิพื้นที่แบบวางซ้อนกัน (Stacked Area Chart)

(ที่มา : http://www.somkiat.cc/wp-content/uploads/2014/12/Where-people-in-Idaho-were-born-620x321.png)

จากรูปที่ 2.7 เป็นข้อมูลการอพยพย้ายถิ่นฐานของประชาชนโดยความหนาของเส้นของแต่ละเส้นคือจำนวนของประชากรที่อพยพ โดย



รูป 2.8 ภาพแสดงการเคลื่อนไหวและการเดินทางของคน

(ที่มา : http://www.somkiat.cc/wp-content/uploads/2014/12/Cultural-history-620x352.png)

จากรูป 2.8 แสดงข้อมูลของคนตั้งแต่ สถานที่เกิด การเดินทาง และสถานที่ตาย มาวาดกราฟเพื่อให้เห็นว่าแต่ละคนมีการเคลื่อนไหว และ เดินทางอย่างไร แสดงถึงการเดินทางของวัฒนธรรมต่างๆ

จากตัวอย่างทั้งหมดนั้นจะเห็นว่า การแสดงผลข้อมูล ไม่กล่าวถึงการได้มาซึ่งกราฟต่างๆ แต่กำลังบอกว่าการทำ การแสดงผลข้อมูล คือ การสื่อความหมาย ซึ่งในงานของ การแสดงผลข้อมูลก่อนที่จะนำเข้ามาสู่การแสดงผลของกราฟหรือสิ่งแสดงผลต่างๆ จะต้องทราบเรื่องต่างๆ ดังต่อไปนี้

2.8.1 ประเภทของข้อมูล (Types of Data)

- ข้อมูลเชิงปริมาณ (Quantitative Data) เป็นข้อมูลประเภทที่สามารถวัดค่าได้ โดยเขียนให้อยู่ในรูปตัวเลขที่สามารถนำไปคำนวนต่อได้ ซึ่งระยะห่างระหว่างช่วงมีความหมาย ตัวอย่างเช่น ตัวเลข ความสูง อุณหภูมิ อายุ เวลา เป็นต้น
- ข้อมูลเชิงกลุ่ม (Categorical Data) เป็นข้อมูลที่แบ่งกลุ่มแยกประเภทอย่างชัดเจน ไม่สามารถนำมาคำนวณทางคณิตศาสตร์ได้ ตัวอย่างเช่น เพศ การศึกษา กลุ่มอายุ สี เป็นต้น
- ข้อมูลเชิงลำดับ (Ordinal Data) เป็นข้อมูลที่รวมกันระหว่าง การแบ่งเชิงกลุ่มและตัวเลขแสดงคุณภาพ โดยเป็นข้อมูลที่จัดเป็นกลุ่ม ซึ่งอันดับของกลุ่มมีความสำคัญ โดยตัวอย่างที่พบกันมากที่สุด คือ ข้อมูลที่ได้จาก แบบสอบถาม เช่น คะแนนความพึงพอใจ จาก 0-5 โดย 0 คือ ไม่พอใจ5 คือ พอใจอย่างมาก ซึ่งคะแนนจากระยะห่าง

ระหว่างช่วงแต่ละกลุ่มไม่มีความหมาย เช่น 1-2 และ 3-4 ซึ่งความรู้สึกที่เพิ่มขึ้นไม่จำเป็นต้องเท่ากัน

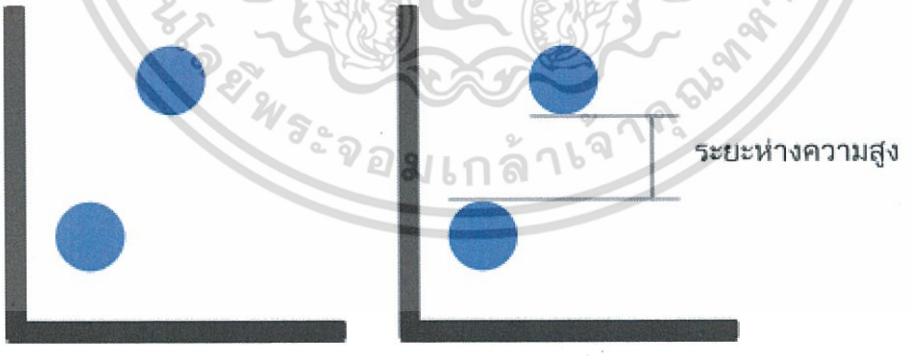
2.8.2 การรับรู้ทางสายตา (Visual Perception)

คือ ความสามารถในการแปลผลของข้อมูล ทราบและรู้ความหมายของสิ่งที่เห็น สิ่งที่มนุษย์สามารถแยกได้ผ่านสายตาได้อย่างง่ายดาย โดยที่ไม่ต้องคิดหรือประมวลผลนั่นมีอยู่ด้วยกัน 3 อย่างด้วยกัน คือ

- ขนาด (size) สามารถเทียบได้กับ ความกว้าง ความยาว ความหนา ความลึก พื้นที่ ตัวอย่างที่เข้าใจง่ายที่สุดดังรูปที่ 2.9

รูป 2.9 ขนาดของวงกลมซ้ายเล็กกว่าวงกลมขวา

- ตำแหน่ง (position) การใช้ตำแหน่งสามารถบอกได้ว่าเป็นการลำดับ เช่น การใช้ความสูงบอกตำแหน่งที่เยอะกว่า ดังรูปที่ 2.10



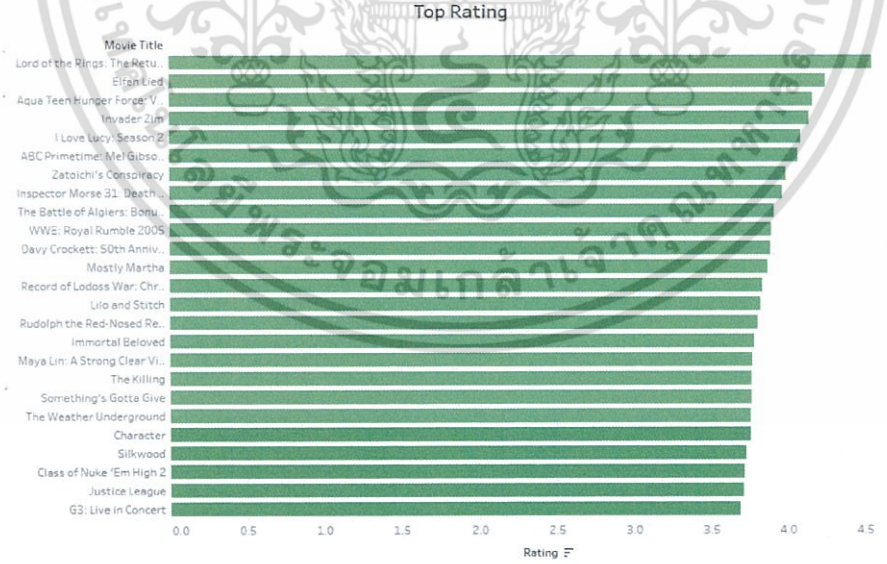
รูป 2.10 วงกลมที่วางตำแหน่งไม่เท่ากันแต่รู้ว่าอันไหนอยู่ลำดับสูงหรือต่ำกว่า

- สี (color) เช่น สีต่างกัน สีเหมือนกัน การไล่สีความสว่าง ดังรูปที่ 2.11

รูป 2.11 แสดงความสามารถของสีที่ช่วยในการแปลผลของข้อมูล

2.8.3 การเลือกประเภทของกราฟที่จะถูกนำมาแสดงผล

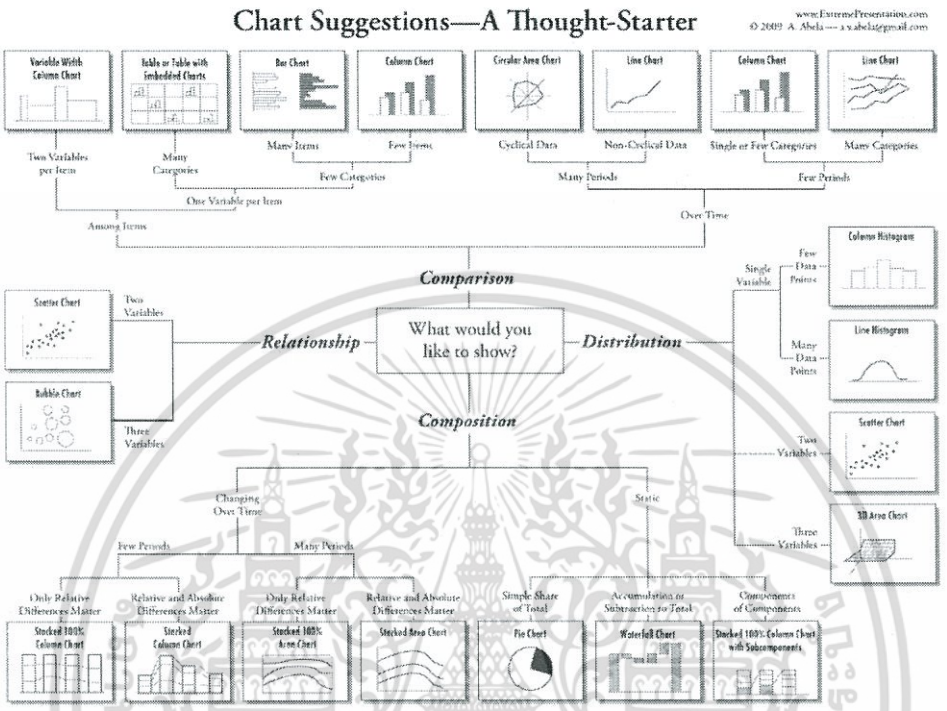
ในการเลือกประเภทของกราฟหรือสื่อที่จะนำมาแสดงผลนั้น จะต้องพิจารณาว่าต้องการให้ผู้ที่รับสาร จะได้รับอะไร ตัวอย่างเช่น ต้องการเปรียบเทียบข้อมูลของภาพยนตร์ให้ผู้รับสารได้ทราบว่า ปีนี้มีภาพยนตร์ใดทำรายสูงสุด หรือจัดลำดับความนิยมจากคะแนนจากคนดูและนักวิจารณ์อย่างไร ซึ่งการเปรียบเทียบนั้นมีได้หลายวิธี อาจจะใช้กราฟแท่ง กราฟเส้น หรือตาราง ในกรณีที่มีภาพยนตร์หลายเรื่อง หรือกรณีที่มีจำนวนภาพยนตร์ไม่มากอย่าง 2-10 เรื่อง ในการเปรียบเทียบอาจจะใช้ กราฟวงกลม แบ่งเป็นส่วนๆ เพื่อบอกความแตกต่างของภาพยนตร์แต่ละเรื่องได้



รูป 2.12 แสดงการจัดลำดับของภาพยนตร์ที่มีคะแนนสูงสุด

จากรูปที่ 2.12 สามารถเห็นได้ทันทีว่าภาพยนตร์ใดที่มีคะแนนสูงสุดและ ต่ำสุด โดยการเปรียบเทียบความยาวของแท่ง

แล้วอะไรบ้างที่สามารถนำไปโชว์ได้ โดยหลักๆ จะมีอยู่ด้วยกัน 4 อย่าง คือ การเปรียบเทียบ (Comparisons) ความสัมพันธ์ (Relationships) การกระจาย (Distribution) การดูองค์ประกอบ (Composition) ซึ่งสามารถสรุปการเลือกกราฟได้จากแผ่นภาพดังรูปที่ 2.13



รูป 2.13 แผนภาพแสดงการเลือกประเภทของกราฟที่จะนำมาแสดง

(ที่มา : http://extremepresentation.typepad.com/files/choosing-a-good-chart-09.pdf)

2.10 เทคโนโลยีที่ใช้

เทคโนโลยีที่ใช้ประกอบไปด้วยภาษาและ framework ในการช่วยพัฒนา ซึ่งทั้งหมดเป็น OpenSource เพื่อให้เหมาะสมต่อการพัฒนาตามวัตถุประสงค์ภายในระยะเวลาที่กำหนด และมีชุมชนของนักพัฒนาที่สามารถให้คำปรึกษาและอ้างอิงได้ โดยมีรายการของเทคโนโลยีที่ใช้ ดังนี้

2.10.1 Python

รูป 2.14 โลโก้ของภาษา Python

ภาษา Python เป็นภาษาโปรแกรมระดับสูง (High-Level Programming Language) สร้างขึ้นในปี 1991 โดย Guido van Rossum สามารถรันภาษา Python ได้ทุก platform และเป็น OpenSource ทำให้มีคนเข้ามาช่วยกันพัฒนาให้ Python มีความสามารถสูงขึ้นและครอบคลุม ทุกลักษณะงาน code ภาษา Python ถูกสร้างขึ้นมาจากภาษา C การประมวลผลทีละบรรทัด (Intepreter) โดยมีจุดเด่นดังนี้

- รองรับการเขียนแบบ โปรแกรมเชิงวัตถุ (OOP : Object Oriented Programming)
- เป็น Dynamic Typing สามารถเปลี่ยนชนิดของตัวแปรได้
- มี Built-in Objects Types คือ โครงสร้างของข้อมูลที่สามารถใช้ได้ใน Python มี list,
- dictionary, string ซึ่งง่ายต่อการพัฒนา
- มี Build-in function ที่สะดวกต่อการเขียนโปรแกรมมากมายเช่น max, min, sorted
- สามารถนำ code ภาษา Python ไปรันภายใต้ภาษา C/C++ ได้
- มี ไลบรารี ที่สนับสนุนงานด้านต่างๆอย่างมากมาย เช่นด้านวิทยาศาสตร์ มี numpy, Scipy,
- scikit-learn ด้านการประมวลผลภาพ มี Opencv ด้าน GUI มี Tkinter, wxPython

2.10.1 JavaScript

รูป 2.15 โลโก้ภาษา JavaScript

JavaScript เป็นภาษาที่ได้รับความนิยมในการสร้างเว็บแอปพลิเคชั่น เพราะปัจจุบันเว็บต่างๆ มีการนำ JavaScript ฝั่งอยู่ในเว็บ โดยมีโปรโตไทพ มาจากภาษา C และอยู่ในฝั่งการทำงานแบบ Client และสามารถทำงานฝั่ง Server-Side โดยใช้ node.js ซึ่งเขียนจากภาษา JavaScript ทั่วไปแล้วโปรแกรมเรียกดูเว็บจะรองรับสำหรับการทำงานของ JavaScript โดยจะต้องตามมาตรฐานของ European Computer Manufactures Association (ECMA) ปัจจุบัน ได้เดินทางมาถึงเวอร์ชั่น ES2017 (ES8)

2.10.2 Django

รูป 2.16 โลโก้ของ Django

Django เป็น open source web framework ที่เขียนด้วยภาษา Python ที่พัฒนา website ได้ง่ายและรวดเร็ว สามารถใช้ RESTful API ในการติดต่อกับ web application ได้

2.10.3 MongoDB

รูป 2.17 โลโก้ของ MongoDB

MongoDB เป็น open source document database โดยเป็นฐานข้อมูลแบบ NoSQL คือไม่มีความสัมพันธ์ และเก็บข้อมูลเป็น JSON (JavaScript Object Notation) ในแต่ละ record ของ database จะเรียกว่า document ซึ่งมีค่า key และ value ตามรูปแบบของ JSON

2.10.4 NLTK

NLTK (Natural Language Toolkit) เป็น ไลบรารี ภาษา Python ของการประมวลผลภาษาธรรมชาติ ซึ่งช่วยในการตัดคำ (Word Tokenize) การหารากศัพท์ของคำ (Stemming and Lemmatization) หาคำพ้องความหมาย (synonyms) โดยมีฐานข้อมูลของคำที่เรียกว่า wordnet

2.10.5 D3.js

รูป 2.18 โลโก้ของ D3.js

เป็น JavaScript ไลบรารี สำหรับสร้าง Data Visualization โดย D3 ย่อมาจากคำว่า Data-Driven Document ที่ช่วยนำข้อมูลไปแสดงผลด้วยกราฟฟิกโดยใช้ HTML, SVG และ CSS

2.10.6 Redis

รูป 2.19 โลโก้ของ Redis

เป็น In-memory data structure store ใช้เก็บข้อมูลบนหน่วยความจำโดยใช้โครงสร้างแบบ Key Value หรือ NoSQL ซึ่งประสิทธิภาพในการ Read, Write, Delete เร็วกว่า Database ที่เก็บบน Hard Disk

2.10.7 scikit-learn

รูป 2.20 โลโก้ของ scikits-learn

เป็น Open Source ไลบรารี ยอดนิยมสำหรับการทำ แมชชีนเลิร์นนิ่ง ด้วยภาษา python ซึ่ง รวบรวมโมเดลทั้ง classification, regression และ clustering รวมถึงการทำ preprocessing และกระบวนการคัดเลือกคุณลักษณะ

2.10.8 React

รูป 2.21 โลโก้ของ React

เป็น JavaScript ไลบรารี สำหรับการพัฒนาเว็บไซต์ที่เป็นส่วนหน้าตาของเว็บไซต์ ที่ได้รับความนิยม เพราะสามารถที่จะสร้างกระบวนการทำงานต่างๆ ที่อยากให้เกิดบนหน้าเว็บได้อย่างสะดวกยิ่งขึ้น โดยใช้ JavaScript es6 และ JSX ซึ่งเป็นมาตรฐานใหม่สำหรับการเขียน JavaScript และ React จะแบ่งส่วนต่างๆ ของหน้าเว็บ และใช้ State ในการเข้าควบคุมกิจกรรมบนหน้าเว็บ

2.10.9 Rechart

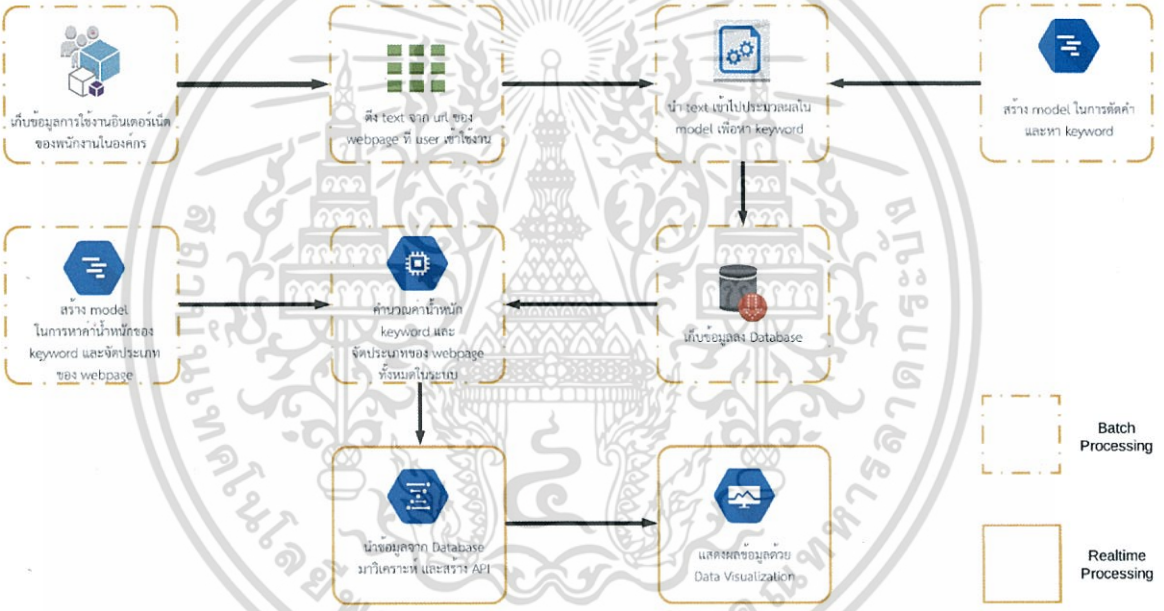
รูป 2.22 โลโก้ของ Recharts

เป็น React Component ไลบรารี (ส่วนประกอบ) ที่ช่วยในการสร้างกราฟสำหรับ React โดยมีส่วน Core หลักเป็น D3.js ซึ่งเป็น JavaScript ไลบรารีสำหรับสร้างกราฟ ผ่าน Html SVG และ CSS

บทที่ 3
การออกแบบและพัฒนา

3.1 ภาพรวมของระบบ

การพัฒนาระบบจะมีส่วนจะแบ่งเป็น 2 ส่วนได้แก่ Batch Processing คือ ส่วนที่เป็นการประมวลผลแบบออฟไลน์ซึ่งก็คือการทำความสะอาดข้อมูลและการแปลงข้อมูลให้อยู่ในรูปแบบที่พร้อมใช้ คือการหาคำสำคัญให้เว็บเพจ หลังจากนั้นก็จะเป็นส่วนของ Realtime Processing คือส่วนนี้เป็นการประมวลผลแบบออนไลน์ซึ่งหลักๆก็คือการวิเคราะห์ข้อมูล และการแสดงผลข้อมูลหัวข้อที่พนักงานสนใจ โดยภาพรวมทั้งหมดสามารถเขียนได้ดังรูปที่ 3.1



รูป 3.1 องค์ประกอบของภาพรวมของระบบ

3.2 ข้อมูลการใช้งานอินเทอร์เน็ตของพนักงาน

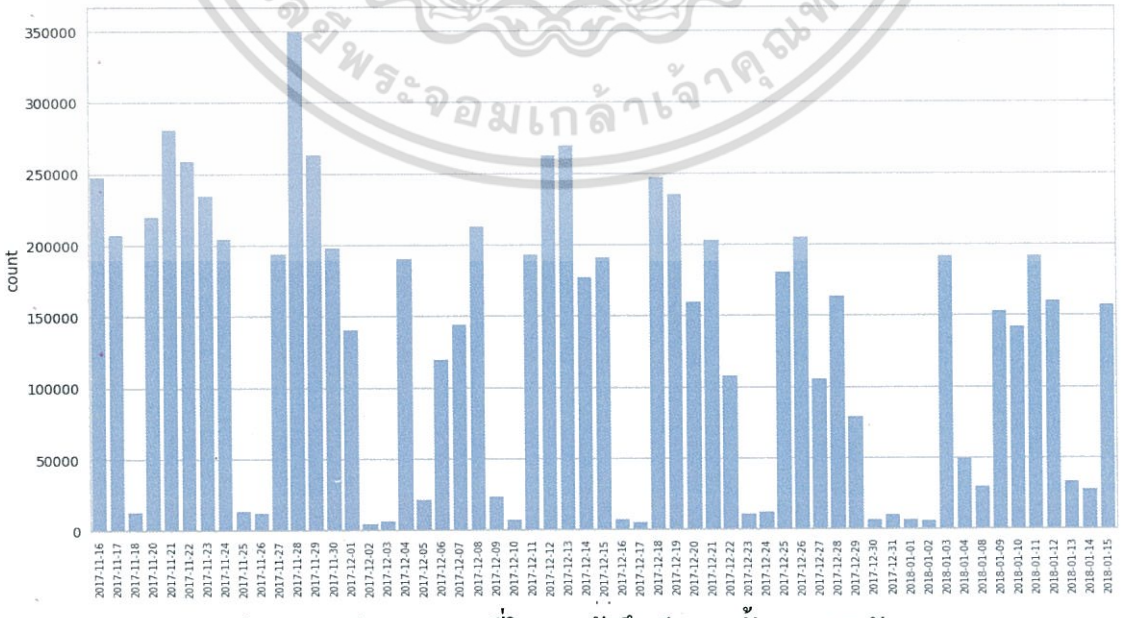
	A	B	C	D	E	F	G	H	I	J	K	L	M
1	eventtime	time_taken	c_ip	cs_username	cs_referer	sc_status	cs_method	rs_content_type	cs_host	cs_bytes	os	name	device
2	9/25/2017	57	172.20.120.205	25cf3371ac7eafaa0f2c79edc2f	-	200	GET	text/html	captive.apple.com	179	Other	Other	Other
3	9/25/2017	82	172.20.34.85	fe7d925ff4a3ee9cc2443a6f862	http://www.ais.co.th/	200	GET	text/html	www.ais.co.th	2063	Windows 8.1	IE	Other
4	9/25/2017	9	172.20.34.85	fe7d925ff4a3ee9cc2443a6f862	-	200	GET	text/html	www.ais.co.th	1580	Windows 8.1	IE	Other
5	9/25/2017	24	172.20.34.85	fe7d925ff4a3ee9cc2443a6f862	http://www.ais.co.th/index.html?int	200	GET	text/html	www.ais.co.th	1400	Windows 8.1	IE	Other
6	9/25/2017	81	172.20.120.167	85bcf1f0eecb08fe5d4adebe72	http://www.seriesyou.com/play/55(	302	GET	text/html	match.adsrvr.org	3024	Windows 8.1	Chrome	Other
7	9/25/2017	305	172.20.120.167	85bcf1f0eecb08fe5d4adebe72	http://www.seriesyou.com/play/55(	200	GET	text/html	synch.optimatic.com	994	Windows 8.1	Chrome	Other
8	9/25/2017	27	172.20.34.85	fe7d925ff4a3ee9cc2443a6f862	-	200	GET	text/html	www.ais.co.th	1000	Windows 8.1	IE	Other
9	9/25/2017	54	172.20.34.85	fe7d925ff4a3ee9cc2443a6f862	http://www.ais.co.th/	200	GET	text/html	fast.ais.demdex.net	1277	Windows 8.1	IE	Other
10	9/25/2017	93	172.20.34.85	fe7d925ff4a3ee9cc2443a6f862	http://www.ais.co.th/index.html?int	200	GET	text/html	www.ais.co.th	1951	Windows 8.1	IE	Other
11	9/25/2017	58	172.20.34.85	fe7d925ff4a3ee9cc2443a6f862	http://www.ais.co.th/index.html?int	200	GET	text/html	www.ais.co.th	1881	Windows 8.1	IE	Other
12	9/25/2017	7	10.20.4.52	af6a8ec5c976a99fa914520d54	http://www.paiduaykan.com/travel/	404	GET	text/html	www.paiduaykan.com	664	Windows 7	IE	Other
13	9/25/2017	12	10.20.4.57	063fc5b94e0614b1c7137be331	-	404	GET	text/html	weather.tile.appex.bing.co	318	Other	Other	Other
14	9/25/2017	8	10.20.4.57	063fc5b94e0614b1c7137be331	-	404	GET	text/html	weather.tile.appex.bing.co	320	Other	Other	Other
15	9/25/2017	8	10.20.4.57	063fc5b94e0614b1c7137be331	-	404	GET	text/html	weather.tile.appex.bing.co	330	Other	Other	Other
16	9/25/2017	286	10.20.4.52	af6a8ec5c976a99fa914520d54	http://www.paiduaykan.com/travel/	200	GET	text/html	www.paiduaykan.com	674	Windows 7	IE	Other
17	9/25/2017	7	10.20.13.22	a19a118e492b223921fe98864	-	404	GET	text/html	weather.tile.appex.bing.co	330	Other	Other	Other
18	9/25/2017	21	172.20.120.140	6ed29e6b111b0384787ec8196	http://www.taladrod.com/w30/iSch	404	GET	text/html	www.taladrod.com	626	Windows 8.1	Chrome	Other
19	9/25/2017	66	172.20.36.91	669baf631c0e9e8b7c38b187b2	http://www.ddproperty.com/%E0%E	404	GET	text/html	tags.crwdcntrl.net	2272	Windows 8.1	Chrome	Other
20	9/25/2017	602	172.20.120.167	85bcf1f0eecb08fe5d4adebe72	http://www.seriesyou.com/play/55(	200	GET	text/html	synch.optimatic.com	994	Windows 8.1	Chrome	Other
21	9/25/2017	94	10.20.120.22	f35444b259fdfad84fd98d142e	https://outlook.live.com/	404	GET	text/html	www.citibank.co.th	500	Windows 7	Chrome	Other
22	9/25/2017	479	172.20.120.175	c5f072ac00b72e43b1c857a433	http://localhost:8080/DevAutobot/V	301	GET	text/html	rawgithub.com	578	Windows 10	Chrome	Other
23	9/25/2017	508	172.20.120.175	c5f072ac00b72e43b1c857a433	http://localhost:8080/DevAutobot/V	301	GET	text/html	rawgithub.com	562	Windows 10	Chrome	Other
24	9/25/2017	358	172.20.120.167	85bcf1f0eecb08fe5d4adebe72	http://www.seriesyou.com/play/55(	302	GET	text/html	altitude.tex-sync.rockyou.n	567	Windows 8.1	Chrome	Other
25	9/25/2017	372	172.20.120.167	85bcf1f0eecb08fe5d4adebe72	http://www.seriesyou.com/play/55(	302	GET	text/html	sync.rhythmxchange.com	449	Windows 8.1	Chrome	Other
26	9/25/2017	40	172.20.120.167	85bcf1f0eecb08fe5d4adebe72	http://www.seriesyou.com/play/55(	302	GET	text/html	match.adsrvr.org	3026	Windows 8.1	Chrome	Other
27	9/25/2017	88	10.20.5.55	8433dbdba837ced31e9c1d9cb	-	200	GET	text/html	www.flgupload.com	436	Windows 8	Chrome	Other
28	9/25/2017	128	172.20.120.167	85bcf1f0eecb08fe5d4adebe72	http://www.seriesyou.com/play/55(	200	GET	text/html	ssum.casalemedia.com	2203	Windows 8.1	Chrome	Other
29	9/25/2017	422	10.20.120.44	80f8487efb44b20bb75f42aedd	-	304	GET	text/html	docs.autodesk.com	488	Windows 7	Safari	Other

รูป 3.2 ตัวอย่างข้อมูลการใช้งาน อินเทอร์เน็ต ของพนักงานในองค์กร

จากข้อมูลตัวอย่าง มีคอลัมน์ดังนี้

- eventtime คือ วัน เวลาที่เข้าเว็บเพจ
- time_taken คือ เวลาที่ใช้ในเว็บเพจ
- c_ip คือ ไอพีแอดเดรสที่เรียกเว็บเพจ
- cs_username คือ ชื่อพนักงานที่ผ่านการแฮช
- cs_referer คือ url ของเว็บเพจ
- cs_method คือ HTTP method ของเว็บเพจ
- rs_content_type คือ ประเภทของเนื้อหาเว็บเพจ
- sc_status คือ สถานะของการตอบรับเว็บเพจ
- cs_host คือ โดเมนของเว็บเพจ
- cs_bytes คือ ขนาดของข้อมูลเว็บเพจ
- os คือ ระบบปฏิบัติการ
- name คือ ชื่อโปรแกรมที่ใช้เข้าเว็บเพจ
- device คือ อุปกรณ์ที่ใช้เข้าเว็บเพจ

มีข้อมูลทั้งหมดประมาณ 300,000,000 แถว หลังจากคัดกรองแล้วเหลือข้อมูลประมาณ 7,500,000 แถวเมื่อทำการจัดกลุ่มข้อมูลแล้ว มีพนักงานทั้งหมด 944 คน และข้อมูลถูกเก็บทั้งหมด 57 วัน ตั้งแต่วันที่ 16 พฤศจิกายน 2560 จนถึงวันที่ 15 มกราคม 2561 ซึ่งมีจำนวนเว็บเพจที่เข้าถึงเฉลี่ยวันละประมาณ 130,000 เว็บเพจ ดังรูปที่ 3.3



รูป 3.3 กราฟแสดงความถี่ในการเข้าถึงเว็บเพจทั้งหมด 57 วัน

3.3 ขั้นตอนการทำงานที่อยู่ในส่วนของ Batch Processing

ขั้นตอนการทำงานในส่วนของ batch processing ที่พัฒนาประกอบไปด้วย 4 ขั้นตอน ดังต่อไปนี้

- **ขั้นตอนที่ 1**

สร้างโมเดลในการตัดคำและหาคำสำคัญและการหารากศัพท์ของคำ โดยวิธีการคือจากอินพุต URL ของเว็บเพจที่ส่งเข้ามา ไปดึงข้อมูลเนื้อหาบนเว็บเพจด้วยฟังก์ชัน get request ของภาษาเพื่อเอาผลลัพธ์ออกมาในรูปแบบของ HTML แล้วคัดกรองแท็กส่วนที่เป็น meta, script style ออกไปเพราะส่วนนี้ไม่ใช่ส่วนที่เป็นเนื้อหาของเว็บเพจ จากนั้นแยกดึงข้อความออกมาจากแท็กที่เป็นส่วนเนื้อหาของเว็บเพจทั้งหมด ได้แก่ title, h1, h2, h3, h4, h5, h6 ที่เป็นส่วนของหัวข้อ (head) และ p ที่เป็นส่วนของเนื้อหา (content) ด้วยไลบรารี BeautifulSoup จากนั้นวนรอบเข้าไปแต่ละประโยคแล้วตรวจจับประโยคที่เข้ามาว่าเป็นภาษาไทยหรือภาษาอังกฤษด้วยช่วงของยูนิโค้ด (unicode) แล้วจึงแยกไปตัดคำตามอัลกอริทึมของแต่ละภาษาดังนี้

ภาษาไทย ในการตัดคำภาษาไทย ใช้ไลบรารี Deepcut ซึ่งใช้อัลกอริทึม คอนโวลูชันแนลนิวรอน เนตเวิร์ค (Convolutional Neural Network) ใน การตัดคำด้วยการตรวจสอบว่าตัวอักษรเป็นตัวเริ่มต้นของคำหรือไม่

ภาษาอังกฤษ อัลกอริทึมที่ใช้ตัดคำภาษาอังกฤษคือ การตัดแต่ละคำในประโยคโดยใช้ช่องว่างและจุดในการแบ่งแยกแต่ละคำ และคัดแยกส่วนที่เป็น คำหยุด ออกจากนั้นนำไปหาคำที่มีรากศัพท์เดียวกัน และพ้องความหมายโดยใช้ เวิร์ดเน็ต ซึ่งใน เวิร์ดเน็ต จะมี dictionary ที่เกี่ยวกับรากศัพท์ของคำ (Stemming and Lemmatizer) ซึ่งจะช่วยให้คำที่มีรากศัพท์เดียวกันกลายเป็นคำเดียวกันและทำให้ประสิทธิภาพของค่าน้ำหนักดีขึ้น

จากนั้นคัดคำหยุด (คำหยุดs) ออกไปแล้วเพิ่มคำนั้นในรายการ ถ้าในรายการมีคำดังกล่าวอยู่แล้ว จะเพิ่มที่ตัวนับ (count) ของคำนั้นแทน และบอกว่าคำนั้นเป็นส่วนหนึ่งของหัวข้อ (head) หรือไม่ จากนั้นคำนวนหาค่า TF ให้แต่ละคำสำคัญด้วยวิธี ดับเบิ้ลนอร์มัลไลเซชัน (Double Normalization) เนื่องจากในแต่ละเว็บเพจมีจำนวนคำไม่เท่ากันและแตกต่างกันมาก วิธีนี้จะช่วยให้จำนวนคำไม่ส่งผลต่อค่า TF โครงสร้างข้อมูลของการเก็บคำสำคัญเป็นดังรูปที่ 3.4 ดังนี้

```
{
  "keyword_1": {
    "tf": "float",
    "idf": "float",
    "weight": "float",
    "in_head": "boolean"
  },
  "keyword_2": {
    "tf": "float",
    "idf": "float",
    "weight": "float",
    "in_head": "boolean"
  }
}
```

รูป 3.4 โครงสร้างข้อมูลของคำสำคัญ

- **ขั้นตอนที่ 2**

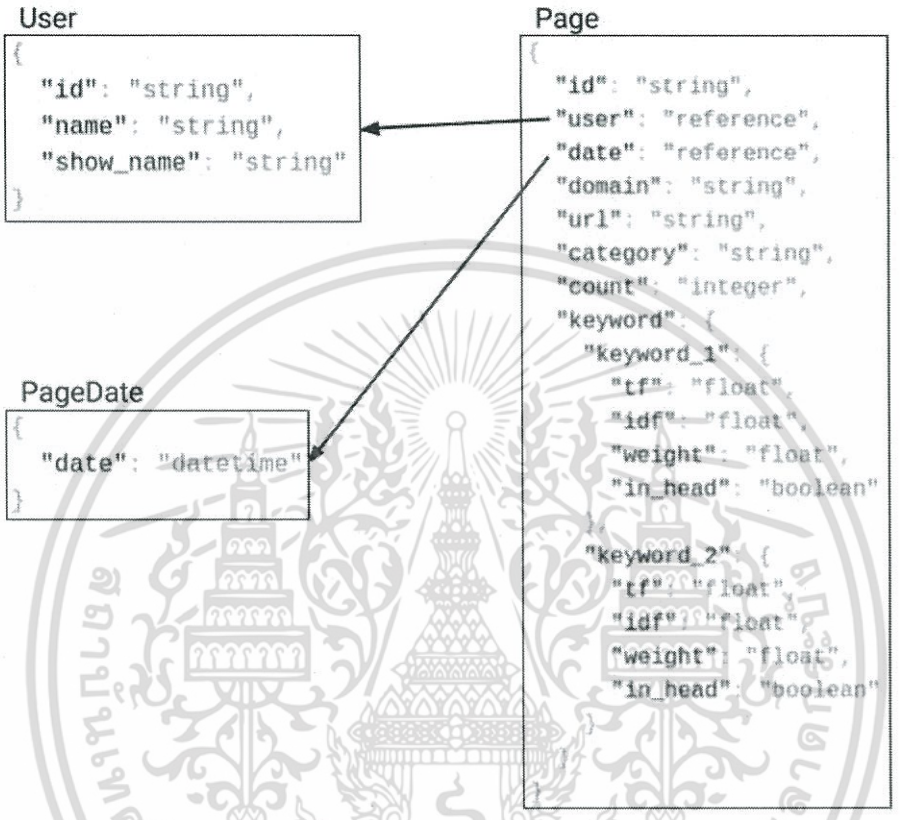
นำ url ของข้อมูลการใช้งาน อินเทอร์เน็ตของพนักงานในองค์กร ในรูปแบบของไฟล์ csv ด้วยไลบรารี่ pandas DataFrame ซึ่งเหมาะกับข้อมูลประเภทตาราง จากนั้นทำการคัดเลือกข้อมูลบางคอลัมน์ที่จำเป็นมาเก็บเท่านั้น ในที่นี้ได้แก่ eventtime, cs_username, cs_referer, cs_method, rs_content_type, sc_status

ในแต่ละแถวของข้อมูล จะคัดกรองข้อมูลด้วยเงื่อนไขดังนี้

- คัดกรองค่าในคอลัมน์ sc_status ที่อยู่ในกลุ่ม 4xx ออกไป เช่น 400, 401, 403, 404 เพราะเป็นเลขสถานะการตอบกลับของเว็บเพจผิดพลาด
- คัดกรองค่าในคอลัมน์ cs_method ที่ไม่ใช่ GET ออกไป เช่น POST, PATCH, PUT, DELETE เพราะ GET เป็นการส่งคำสั่งไปเพื่อขอข้อมูลเว็บเพจที่เหลือเป็นการส่งคำสั่งไปพร้อมค่าอินพุตแต่ไม่ได้ต้องการขอข้อมูลเว็บเพจ
- คัดกรองค่าในคอลัมน์ rs_content_type ที่เป็น text/html เพราะเป็นประเภทเนื้อหาที่เป็น text
- คัดกรองเว็บเพจที่เกี่ยวกับเครือข่ายสังคมออก (Social Network) เช่น Facebook, Twitter เนื่องจากไม่สามารถนำเนื้อหาจากเว็บเพจออกมาได้ เพราะติดปัญหาเรื่องความเป็นส่วนตัวในเนื้อหาของแต่ละบุคคลซึ่งบุคคลอื่นไม่สามารถเข้าถึงได้

- **ขั้นตอนที่ 3**

หลังจากคัดกรองแล้วนำ URL ที่ได้ไปหาคำสำคัญออกมาแล้ว นำไปเก็บลงดาต้าเบสประเภท NoSQL ที่มีโครงสร้างดังรูปที่ 3.5 ดังนี้



รูป 3.5 โครงสร้าง NoSQL database ในรูปแบบ json

โดยแต่ละแถวของข้อมูลทำการสร้าง Document Page ขึ้นมาเก็บข้อมูลที่ได้จากอินพุต โดยปล่อยให้ category ว่างไว้เพื่อเก็บไว้หาต่อไปและสร้าง document user จากคอลัมน์ cs_username และ pagedate จากคอลัมน์ eventtime แยกไว้เพื่อให้ง่ายต่อการค้นหาและจับกลุ่มเว็บเพจตอนที่สร้าง API สำหรับดึงข้อมูลคำสำคัญไปแสดงผลบนเว็บไซต์ให้รองรับการคัดกรองด้วยวันที่และพนักงาน

จากนั้นการเก็บข้อมูลแถวต่อไปถ้า URL ที่เข้ามาซ้ำกับเงื่อนไขได้แก่ URL, user, date ไม่ต้องเพิ่ม document ที่ URL นั้นอีก ให้เพิ่มตัวนับ (count) ของ document ที่มีเงื่อนไขดังกล่าวแทน และทำให้ไม่ต้องนำ URL เดิมไปหาคำสำคัญซ้ำอีก

อัลกอริทึมที่ใช้ในการจำแนกประเภทของเว็บเพจจะใช้ การเรียนรู้แบบมีผู้สอน (Supervise Learning) ค่าที่นำมาใช้ในการสร้างโมเดลทำนายคือ ค่าน้ำหนักของแต่ละคำสำคัญ โดยแบ่งข้อมูลบางส่วนมาทำการสร้างโมเดลในรูปของเมทริกซ์

เนื่องจากชื่อพนักงานจากข้อมูลเป็นชื่อที่ถูกเข้ารหัส (encrypt) ไว้ทำให้ต้องแต่งชื่อพนักงานขึ้นใหม่เพื่อให้เหมาะแก่การแสดงผลบนหน้าเว็บ

- **ขั้นตอนที่ 4**

หาค่า IDF ให้ทุกคำสำคัญของเว็บเพจทั้งหมดในระบบ โดยนับว่าแต่ละคำสำคัญมีการปรากฏอยู่ทั้งหมดกี่เอกสาร คำสำคัญที่หาค่า IDF ได้แล้วให้เก็บคำสำคัญนั้นพร้อมกับค่า IDF ไว้ในคลังข้อมูลคำศัพท์ และเมื่อวนรอบไปเจอคำสำคัญที่มีอยู่ในคลังข้อมูลคำศัพท์อยู่แล้ว ให้นำค่า IDF จากคำสำคัญนั้นมาเก็บได้เลย เพราะคำสำคัญเดียวกันจะปรากฏอยู่ในจำนวนเอกสารที่เท่ากัน ทำให้คำสำคัญที่ซ้ำกันไม่ต้องคำนวนใหม่ เสร็จแล้วให้หาค่าน้ำหนักของคำสำคัญโดยสูตร TF-IDF

จากนั้นจะจำแนกประเภทของเว็บเพจด้วยการใช้แมชชีนเลิร์นนิ่งในการจำแนกเว็บเพจออกเป็น 2 ประเภท คือ ความรู้ หรือเว็บเพจที่เกี่ยวกับความรู้เช่น เทคโนโลยี เว็บบอร์ด ถามตอบความรู้ ข่าวเทคโนโลยี โดยเน้นไปที่เทคโนโลยีสารสนเทศ และ ทั่วไป หรือเว็บเพจที่เกี่ยวกับเรื่องทั่วไปเช่น ข่าว ขายสินค้าออนไลน์ กีฬา เป็นต้น

อัลกอริทึมที่ใช้ในการจำแนกประเภทของเว็บเพจจะใช้ การเรียนรู้แบบมีผู้สอน ค่าที่นำมาใช้ในการสร้างโมเดลทำนายคือ ค่าน้ำหนักของแต่ละคำสำคัญ โดยแบ่งข้อมูลบางส่วนมาทำการสร้างโมเดลในรูปของเมทริกซ์ดังตัวอย่าง ดังรูปที่ 3.6

page	keyword_1	keyword_2	keyword_3	keyword_4	keyword_5	keyword_6	keyword_7	keyword_8	keyword_9	keyword_n
page_1	9.179064	5.435458	2.560890	3.820813	1.565436	3.817849	5.613395	7.576682	2.980818	5.493478
page_2	4.963212	1.679560	9.205659	4.260140	3.618855	2.437624	8.038618	5.750213	5.632234	1.733572
page_3	3.757745	7.935371	9.287812	8.881634	6.661657	4.673586	6.447918	7.132517	6.465413	5.839657
page_4	7.090121	3.261490	1.725108	1.340839	6.013419	8.471263	6.900003	6.597542	9.549436	8.668056
page_5	2.923616	7.591371	6.157438	6.683121	0.674109	0.825042	4.173485	4.806675	6.963776	4.303043
page_6	3.920800	0.021722	3.168544	3.078493	4.064853	4.546366	6.143346	2.611181	1.079422	2.873634
page_7	1.897281	1.123663	4.289293	1.411454	9.676873	1.250979	8.934107	2.434199	7.666181	8.157815
page_8	4.850514	4.668384	1.094334	2.347322	9.426179	9.158469	9.898994	3.317493	1.650162	4.250561
page_9	7.338420	6.414785	7.766083	7.284632	0.136167	0.281895	2.770251	3.530710	0.527072	0.176906
page_n	7.936185	5.415848	6.199028	5.980616	7.459046	3.930016	1.191609	6.022550	2.286241	5.348198

รูป 3.6 ตัวอย่างการเตรียมโครงสร้างข้อมูลเพื่อนำไปสร้างโมเดลทำนายประเภทของเว็บเพจ

จากนั้นทดสอบความถูกต้องของโมเดลด้วยวิธี k fold cross validation

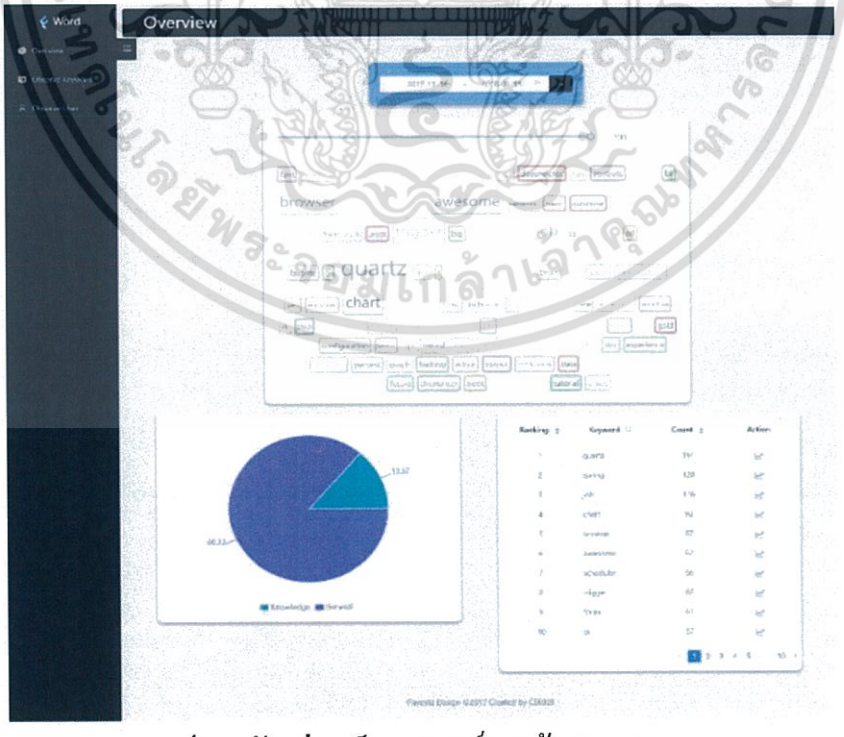
3.4 ขั้นตอนการทำงานที่อยู่ในส่วนของ Realtime Processing

นำเสนอข้อมูลที่วิเคราะห์ออกมาในรูปแบบของเว็บแอปพลิเคชันในลักษณะเป็นหน้า แดชบอร์ด โดยรับส่งข้อมูลกับ เว็บเซิร์ฟเวอร์ (Web Server) ด้วยโปรโตคอล RESTful API ซึ่งมีรูปแบบเป็น JSON ดังรูปที่ 3.7

รูป 3.7 กระบวนการรับส่งข้อมูลระหว่าง Web Dashboard กับ Database

ฝั่ง เว็บเซิร์ฟเวอร์ จะพัฒนา API service ที่เกี่ยวกับการค้นข้อมูลคำสำคัญของเว็บเพจ โดยคำสำคัญที่ถูกนำมาพิจารณาคือคำสำคัญที่อยู่ในแท็กของหัวข้อ (head) เท่านั้น เพราะคำสำคัญที่อยู่ในแท็กหัวข้อทั้งหมดจะสามารถบอกความสำคัญของเว็บเพจนั้นได้มากกว่าคำสำคัญที่อยู่ส่วนอื่นๆ เช่น เนื้อหา (paragraph)

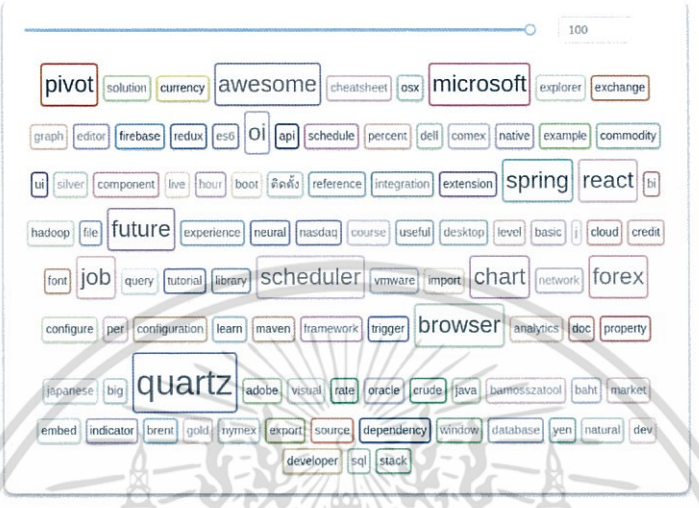
- **หน้า Overview**



รูป 3.8 ตัวอย่าง เว็บแดชบอร์ด หน้า Overview

จากรูปที่ 3.8 เป็นหน้าภาพรวมคำสำคัญของเว็บเพจทั้งหมดของระบบ สามารถคัดกรองด้วยช่วงเวลาได้ แบ่งเป็น 3 ส่วน

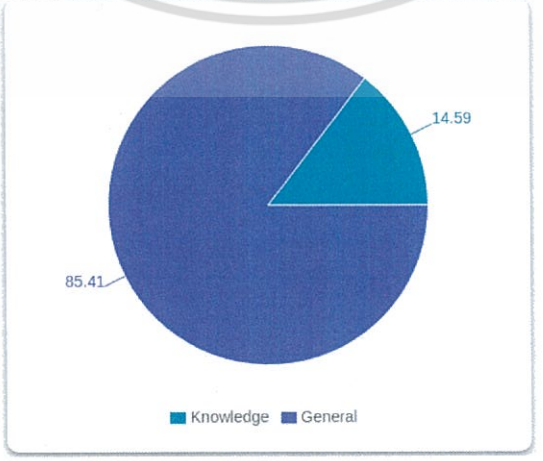
1) คำสำคัญยอดนิยม



รูป 3.9 ตัวอย่างรายการคำสำคัญยอดนิยม

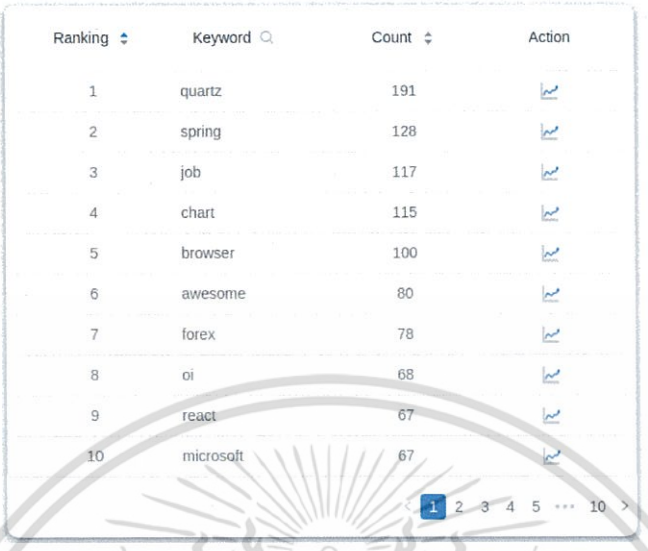
จากรูปที่ 3.9 คำสำคัญยอดนิยมคิดจากผลรวมของค่าน้ำหนักของแต่ละคำสำคัญในเว็บเพจประเภท ความรู้ เท่านั้นเพราะว่าเว็บเพจประเภท ความรู้ ให้ผลลัพธ์คำสำคัญที่เป็นประโยชน์กับองค์กรมากกว่าเว็บเพจประเภททั่วไป โดยตัวอักษรที่มีขนาดใหญ่ที่สุด บ่งบอกว่าคำสำคัญนั้นถูกสนใจโดยพนักงานมากที่สุดในช่วงเวลาที่ถูกคัดกรอง โดยสามารถปรับจำนวนคำสำคัญที่แสดงได้ถึง 100 คำสำคัญ

2) กราฟแผนภูมิวงกลม บอกสัดส่วนของเว็บเพจในระบบทั้งสองประเภท โดยบอกเป็นเปอร์เซ็นต์



รูป 3.10 ตัวอย่างกราฟแผนภูมิวงกลม

3) ตารางแจกแจงความถี่คำสำคัญ



Ranking	Keyword	Count	Action
1	quartz	191	
2	spring	128	
3	job	117	
4	chart	115	
5	browser	100	
6	awesome	80	
7	forex	78	
8	oi	68	
9	react	67	
10	microsoft	67	

1 2 3 4 5 ··· 10 >

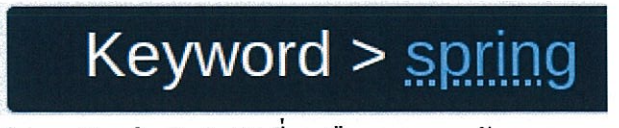
รูป 3.11 ตัวอย่างตารางแจกแจงความถี่คำสำคัญ

จากรูปที่ 3.11 ตารางแจกแจงความถี่คำสำคัญจากแผนภาพคำสำคัญว่ามีความถี่ของคำสำคัญนั้นมากเท่าใด เรียงลำดับจากมากไปน้อย สามารถค้นหาคำสำคัญได้ และแต่ละคำสำคัญสามารถเลือกกดเข้าไปดูรายละเอียดได้ที่หน้าต่อไป

- **หน้า Observe Keyword**

รูป 3.12 ตัวอย่างหน้า Observe Keyword

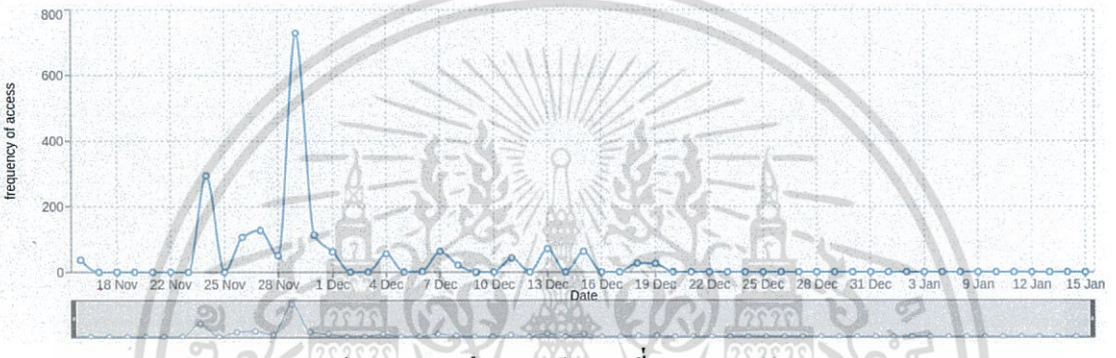
จากรูปที่ 3.12 เป็นหน้าแสดงรายละเอียดของคำสำคัญที่เลือกมาจากหน้า overview โดยจะขึ้นคำสำคัญที่ต้องการสังเกตมุมด้านบน ดังรูปที่ 3.13



รูป 3.13 ตัวอย่างคำสำคัญที่ถูกเลือกมาจากหน้า Overview

โดยรายละเอียดนั้นถูกกรองมาจากเว็บเพจที่อยู่ในประเภท ความรู้ และยังสามารถกรองด้วยช่วงเวลาได้เหมือนดังหน้า Overview ซึ่งสามารถแบ่งออกเป็น 2 ส่วน หลักๆ ด้วยกันคือ

1) กราฟความถี่ของคำสำคัญ



รูป 3.14 ตัวอย่างกราฟความถี่ของคำสำคัญ

จากรูปที่ 3.14 เป็นส่วนที่แสดงความถี่ของการเข้าถึงคำสำคัญนั้นของพนักงานทุกคนในแต่ละวัน

2) ตารางแจกแจงความถี่พนักงานที่เข้าถึงคำสำคัญนั้น

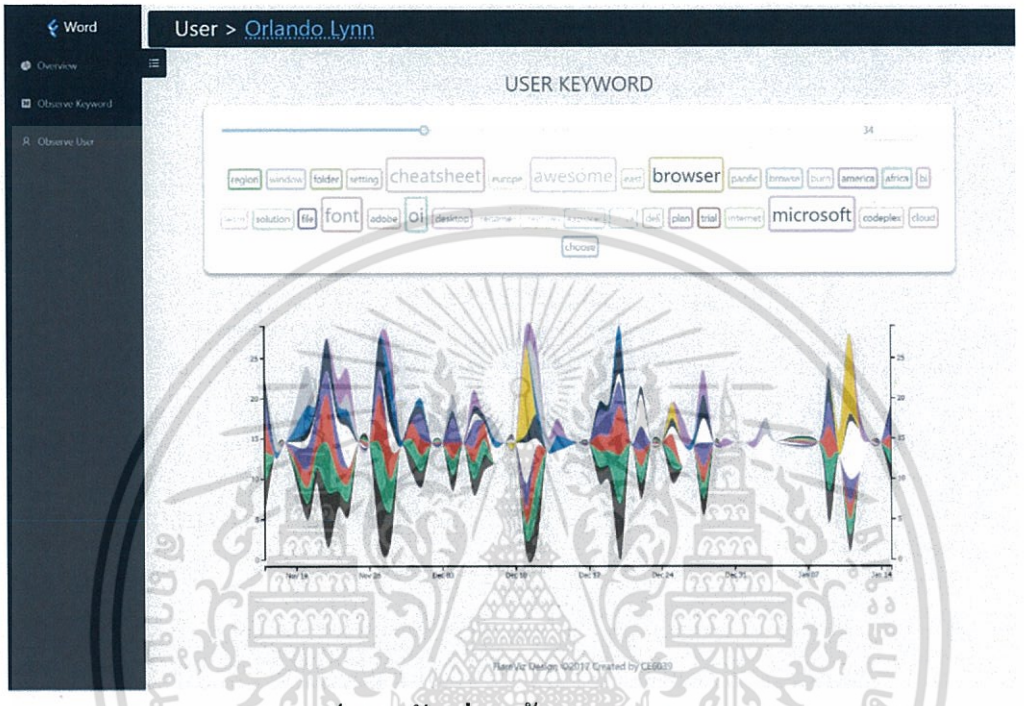
Rank	User Name	Frequency	Action
1	Walter Williamson	435	
2	Sharen Ramsey	207	
3	Trent Carroll	145	
4	Marine Marshall	54	
5	Jacinda Foster	27	

1

รูป 3.15 ตัวอย่างตารางแจกแจงความถี่พนักงานที่เข้าถึงคำสำคัญนั้น

จากรูปที่ 3.15 เป็นตารางที่แสดงความถี่ที่พนักงานแต่ละคนเข้าถึงคำสำคัญนั้นเรียงจากมากไปน้อยตามช่วงเวลาที่ถูกคัดกรอง และสามารถเลือกกดเข้าไปดูรายละเอียดของพนักงานคนนั้นได้ที่หน้าต่อไป

- **หน้า Observe User**

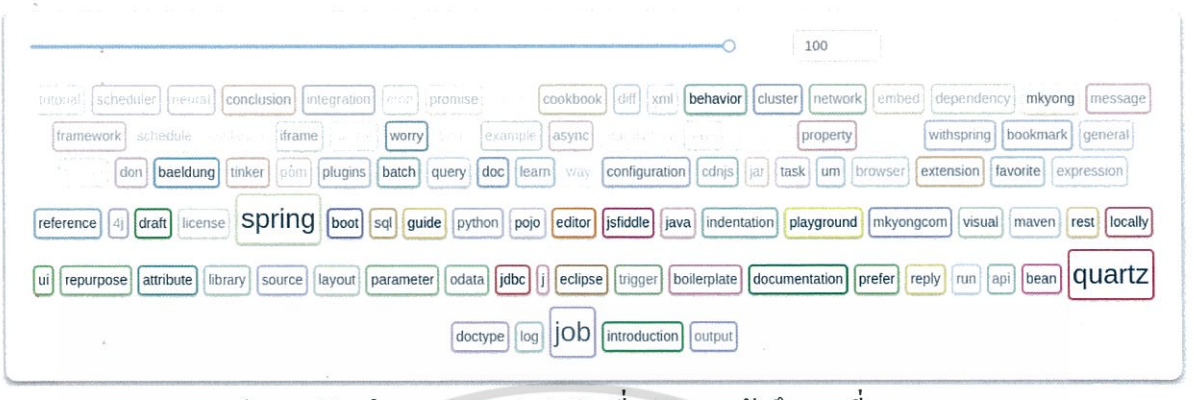


รูป 3.16 ตัวอย่างหน้า Observe User

จากรูปที่ 3.16 เป็นหน้าแสดงรายละเอียดคำสำคัญที่พนักงานหนึ่งคนเข้าถึงทั้งหมดโดยเลือกพนักงานมาจากหน้า Observe Keyword โดยจะแสดงชื่อของพนักงานดังรูปที่ 3.17 และเป็นคำสำคัญที่ถูกกรองมาจากเว็บเพจประเภท ความรู้ แบ่งเป็น 2 ส่วน

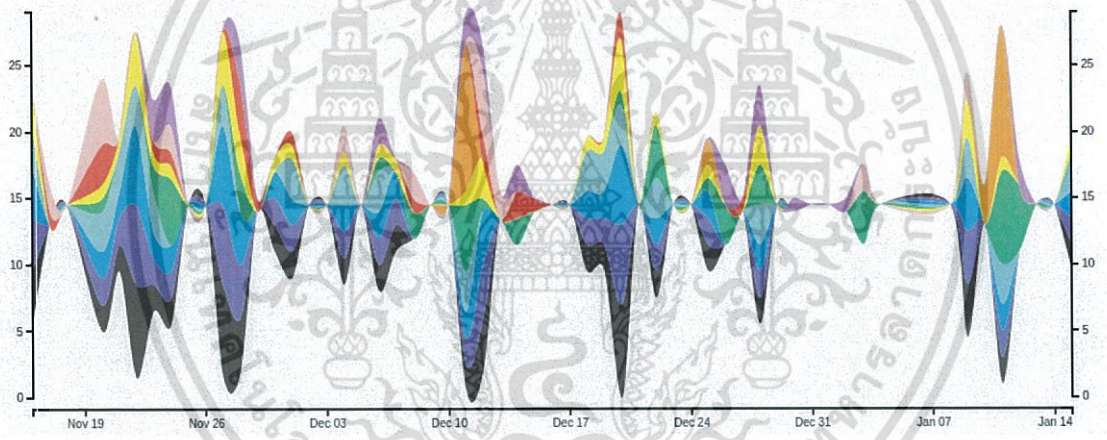
รูป 3.17 ตัวอย่างพนักงานที่ถูกเลือกมาจากหน้า Observe Keyword

1) แผนภาพคำสำคัญที่พนักงานเข้าถึงมากที่สุด



รูป 3.18 ตัวอย่างแผนภาพคำสำคัญที่พนักงานเข้าถึงมากที่สุด

2) กราฟสตรีม (Stream)



รูป 3.19 ตัวอย่างกราฟสตรีม

จากรูปที่ 3.19 เป็นกราฟที่แสดงความต่อเนื่องในการเข้าถึงคำสำคัญในแต่ละวันโดย 1 สีคือ 1 คำสำคัญถ้าสีใดมีความกว้างของกราฟมากแปลว่าคำสำคัญนั้นมีการเข้าถึงมาก โดยแบ่งเป็น 10 สี คือ คำสำคัญที่พนักงานเข้าถึงมากที่สุด 10 คำสำคัญ

บทที่ 4

การทดลองและผลการทดลอง

กระบวนการทั้งหมดในการทำความสะอาดข้อมูล, หาคำสำคัญ, วิเคราะห์ข้อมูล จะใช้ redis ในการช่วยเพิ่มตัวประมวลผลให้เป็นการทำงานแบบขนาน (Parallel) เพื่อเพิ่มประสิทธิภาพในการประมวลผล และใช้การแคช (cache) เพื่อเก็บผลลัพธ์การประมวลผลบนหน่วยความจำทำให้ไม่ต้องประมวลผลซ้ำ เนื่องจากข้อมูลจริงจากองค์การมีขนาดไฟล์ที่ใหญ่ถึง 6 GB โดยใช้เวลาในการพัฒนาส่วนใหญ่ไปกับการทำความสะอาดข้อมูล

คอมพิวเตอร์ที่ใช้ในการประมวลผลใช้ซีพียู Intel Xeon E5-2620 2.40GHz จำนวน 8 core, หน่วยความจำ 16 GB, ฮาร์ดดิสก์ 130 GB

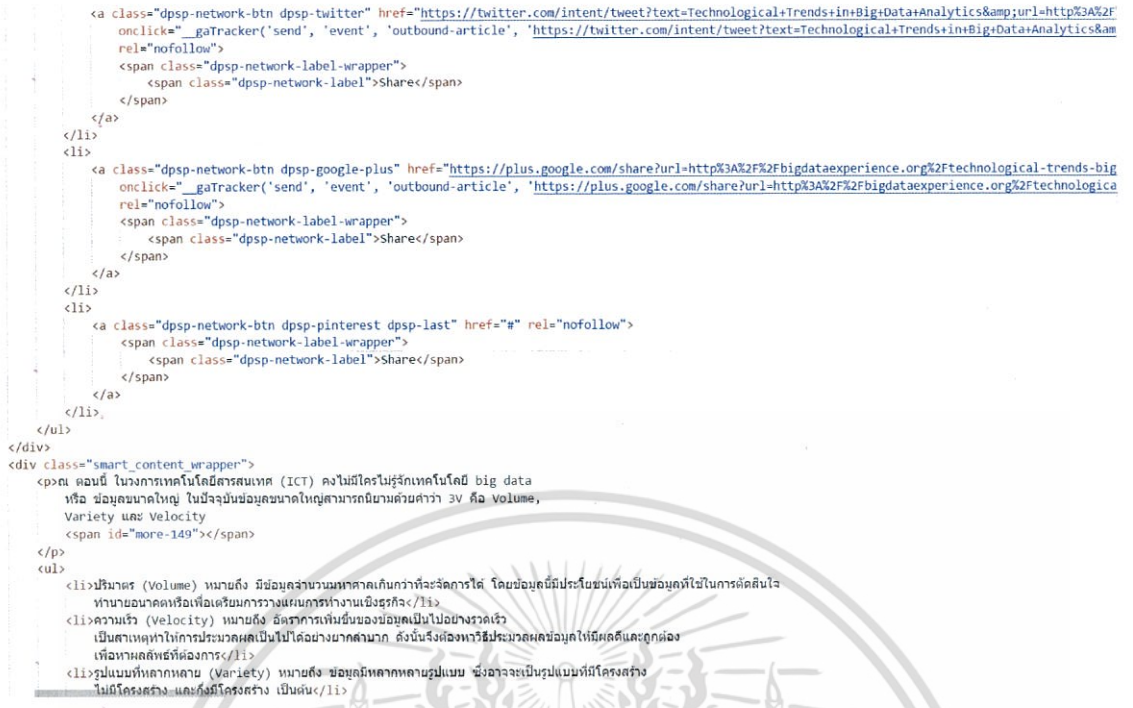
ในเนื้อหาของบทนี้จะเป็นการใช้ข้อมูลการใช้งานอินเทอร์เน็ตของพนักงานในองค์กรทั้งหมด หลังจากกรองเงื่อนไขของคอลัมน์ sc_status, cs_method, rs_content_type แล้วเหลือประมาณ 7,500,000 แถวของไฟล์ .csv จากทั้งหมด 300,000,000 แถว ขนาดไฟล์เหลือ 1.42 GB จากนั้นทำการเก็บข้อมูลเว็บเพจลงดาต้าเบส NoSQL โดยข้อมูลที่มี URL, วันที่ และพนักงานซ้ำ จะไม่ทำการเพิ่ม object ใหม่ลงดาต้าเบส จะใช้การเพิ่มค่าความถี่แทน (count) แทน โดยการเก็บคีย์เวิร์ดนั้นจะทำการเก็บทั้งภาษาไทยและภาษาอังกฤษ

ตัวอย่างเว็บเพจที่ใช้ในการหาคำสำคัญ " "

" http://bigdataexperience.org/technological-trends-big-data-analytics "

4.1 ผลลัพธ์จากการหา คำสำคัญ จาก เว็บเพจ

ทำการดึงข้อมูลข้อความออกมาจากเว็บเพจโดยการ get request ด้วยไลบรารี URLlib3 ไปที่ URL ของเว็บเพจได้ ดังรูปที่ 4.1



รูป 4.1 ผลลัพธ์จากการดึงข้อมูล text ออกมาจาก webpage บางส่วน

จะได้ผลลัพธ์ข้อความออกมาในลักษณะของ html จากนั้นนำไปคัดแท็ก meta, script ซึ่งไม่ใช่ส่วนที่เป็นเนื้อหาออกไป แล้ววนรอบเข้าไปในแต่ละแท็กของผลลัพธ์ ถ้าเจอข้อความภาษาไทยหรือภาษาอังกฤษ โดยแบ่งด้วยค่า unicode ซึ่งภาษาอังกฤษจะอยู่ในช่วง 65 - 122 และภาษาไทยจะอยู่ในช่วง 3585 - 3673 จากนั้นนำไปตัดคำด้วยอัลกอริทึมของแต่ละภาษา และคัดแยกส่วนที่เป็น คำหยุด และอักขระพิเศษออกไป โดยคำที่ซ้ำให้เพิ่มตัวนับของคำนั้น เพื่อนำค่าความถี่ของคำนั้นไปหาค่า TF ในภายหลัง และบอกว่าคำนั้นอยู่ในส่วนของแท็ก head ใน html หรือไม่ ซึ่งได้แก้ title, h1, h2, h3, h4, h5, h6 ซึ่งจะนำมาใช้เป็น คำสำคัญ เพราะคำในส่วนนี้สามารถบอกใจความสำคัญของเว็บเพจนี้ได้ ซึ่งจะได้ผลดังรูปที่ 4.2

```
{
  "count": 744,
  "keyword": {
    "technological": {
      "tf": 4,
      "idf": 0,
      "weight": 0,
      "in_head": "Yes"
    },
    "trend": {
      "tf": 6,
      "idf": 0,
      "weight": 0,
      "in_head": "Yes"
    },
    "big": {
      "tf": 37,
      "idf": 0,
      "weight": 0,
      "in_head": "Yes"
    },
    "data": {
      "tf": 42,
      "idf": 0,
      "weight": 0,
      "in_head": "Yes"
    },
    "analytics": {
      "tf": 10,
      "idf": 0,
      "weight": 0,
      "in_head": "Yes"
    },
```


รูป 4.2 ผลลัพธ์จากการตัดคำและนับคำซ้ำในส่วนของภาษาอังกฤษบางส่วน

```
"เทคโนโลยี": {
  "tf": 8,
  "idf": 0,
  "weight": 0,
  "in_head": "No"
},
"รู้จัก": {
  "tf": 4,
  "idf": 0,
  "weight": 0,
  "in_head": "No"
},
"ข้อมูล": {
  "tf": 79,
  "idf": 0,
  "weight": 0,
  "in_head": "No"
},
"ขนาด": {
  "tf": 20,
  "idf": 0,
  "weight": 0,
  "in_head": "No"
},
"ใหญ่": {
  "tf": 22,
  "idf": 0,
  "weight": 0,
  "in_head": "No"
},
"ปัจจุบัน": {
  "tf": 5,
  "idf": 0,
  "weight": 0,
  "in_head": "No"
},
```


รูป 4.3 ผลลัพธ์จากการตัดคำและนับคำซ้ำในส่วนของภาษาไทยบางส่วน

จากผลลัพธ์รูปที่ 4.3 คำว่า “big”, “data” มีค่าความถี่มากที่สุด ในส่วนของภาษาอังกฤษและคำว่า “ข้อมูล”, “ขนาด”, “ใหญ่” แต่ ในส่วนของภาษาไทย แต่ว่าคำที่จะถูกใช้เป็นคำสำคัญคือคำที่อยู่ในส่วนของภาษาอังกฤษเพราะอยู่ในแท็ก head (“in_head”: “Yes”)

4.2 เปรียบเทียบอัลกอริทึมในการหารากศัพท์ของคำในภาษาอังกฤษ

ตัวอย่างเว็บเพจ https://en.wikipedia.org/wiki/Computer_engineering

ซึ่งเป็นเว็บเพจ ที่เกี่ยวกับข้อมูลของ Computer Engineering

4.2.1 เปรียบเทียบระหว่าง word stemming และ word lemmatization

ผลทดสอบจาก อัลกอริทึม PorterStemmer ได้ผลลัพธ์ดังรูปที่ 4.4

```
{
   "count": 2217,
   "keyword": {
      "engin": 150,
      "comput": 110,
      "technolog": 81,
      "system": 48,
      "softwar": 32,
      "work": 27,
      "design": 26,
      "edit": 25,
      "thi": 24,
      "degre": 23,
      "scienc": 22,
      "develop": 22,
      "hardwar": 20,
      "articl": 19,
      "network": 15,
      "field": 14,
      "includ": 14,
      "templat": 13,
      "electron": 12,
      "commun": 12,
      "main": 12,
      "electr": 11,
      "integr": 11,
      "inform": 11,
```


รูป 4.4 ผลของการหารากศัพท์ของอัลกอริทึม PorterStemmer

ผลทดสอบจาก อัลกอริทึม SnowballStemmer (Porter2) ได้ผลลัพธ์ดังรูปที่ 4.5


```
{
   "count": 2182,
   "keyword": {
      "engin": 150,
      "comput": 110,
      "technolog": 81,
      "system": 48,
      "softwar": 32,
      "work": 27,
      "design": 26,
      "edit": 25,
      "degre": 23,
      "scienc": 22,
      "develop": 22,
      "hardwar": 20,
      "articl": 19,
      "network": 15,
      "field": 14,
      "includ": 14,
      "templat": 13,
      "electron": 12,
      "main": 12,
      "electr": 11,
      "integr": 11,
      "communic": 11,
      "inform": 11,
      "process": 11,
```


รูป 4.5 ผลของการหารากศัพท์ของอัลกอริทึม SnowballStemmer

ผลทดสอบจาก อัลกอริทึม WordNetLemmatizer ได้ผลลัพธ์ดังรูปที่ 4.6


```
{
   "count": 2127,
   "keyword": {
      "engineer": 149,
      "computer": 98,
      "technology": 60,
      "system": 48,
      "software": 32,
      "work": 27,
      "design": 26,
      "edit": 25,
      "degree": 23,
      "science": 22,
      "technological": 21,
      "hardware": 20,
      "article": 19,
      "network": 15,
      "field": 14,
      "include": 14,
      "template": 13,
      "main": 12,
      "electrical": 11,
      "communication": 11,
      "process": 11,
      "program": 11,
      "retrieve": 11,
      "wikipedia": 10,
```


รูป 4.6 ผลของการหารากศัพท์ของอัลกอริทึม WordNetLemmatizer

4.3 ผลลัพธ์จากกการทำความสะอาดข้อมูล

ข้อมูลเว็บเพจที่เก็บบนดาต้าเบสทั้งหมดแล้วจะได้พนักงาน ทั้งหมด 944 คน และวันที่เก็บข้อมูลทั้งหมด 57 วัน ตั้งแต่วันที่ 16 พฤศจิกายน 2560 ถึง 15 มกราคม 2561 เนื่องจากข้อมูลมีจำนวนมาก จึงนำข้อมูลมาวิเคราะห์ต่อเพียงพนักงาน 100 คนซึ่งมีเว็บเพจรวมกันประมาน 49,000 เว็บเพจ จากนั้นทำการสำรวจข้อมูลเว็บเพจที่มีพบว่ามีเว็บเพจที่เกิดการเข้าถึงมากเป็นพิเศษ และจะต้องถูกคัดออกเนื่องจากไม่สามารถให้ประโยชน์กับการวิเคราะห์ข้อมูลได้แก่

- greenwave.fm เป็นเว็บเพจที่เกี่ยวกับฟังวิทยุและไม่มีเนื้อหาบนเว็บเพจ พบว่ามีการเข้าถึงมากเป็นพิเศษรวมกันถึง 10,000 กว่าครั้ง นับจากความถี่ ซึ่งความเป็นจริงแล้วในเวลา 57 วัน มีการเข้าถึงมากเกินไปและไม่เป็นประโยชน์กับการวิเคราะห์ข้อมูล
- เว็บเพจที่ URL ลงท้ายด้วย .css เนื้อหาบนเว็บเพจเป็นโค้ดภาษา css ซึ่งไม่มีเนื้อหาที่เป็นข้อความ
- เว็บเพจที่มีโดเมนเป็นของบริษัทนั้น ซึ่งเป็นเว็บเพจที่พนักงานใช้ในการลงชื่อเข้าใช้อินเทอร์เน็ตในองค์กร ซึ่งไม่ใช่เว็บเพจที่พนักงานเข้าถึงเพราะความสนใจ จึงไม่สามารถนำไปวิเคราะห์ได้

- เว็บเพจที่ URL ลงท้ายด้วย .swf เนื้อหาบนเว็บเพจเป็น adobe flash ซึ่งไม่มีเนื้อหาที่เป็นข้อความ
- เว็บเพจที่มีโดเมนเป็น hao123 เนื่องจากเป็น search engine ของ baidu ซึ่งไม่ใช่เว็บเพจที่เป็นเนื้อหาที่มีประโยชน์ต่อการวิเคราะห์ข้อมูล
- http://www.sanook.com/ เป็นเว็บเพจหน้า home page เกี่ยวกับเรื่องทั่วไป ข่าว ซึ่งไม่ได้เจาะจงว่าเป็นเนื้อหาเกี่ยวกับอะไร จึงไม่มีประโยชน์ต่อการวิเคราะห์ข้อมูล
- เว็บเพจที่ URL ลงท้ายด้วย .js เนื้อหาบนเว็บเพจเป็นโค้ดภาษา javascript ซึ่งไม่มีเนื้อหาที่เป็นข้อความ
- เว็บเพจที่เป็นหน้า home page โดเมน msn เนื่องจากเป็นเว็บเพจที่ถูก redirect มาจากเว็บเพจที่ใช้ยืนยันตัวตนเข้าใช้อินเทอร์เน็ตขององค์กร ถึงไม่ใช่เหตุผลนี้ก็ยังต้องถูกคัดออกเนื่องจากเป็นเว็บเพจที่เป็น home page ที่ไม่เจาะจงว่าเป็นเนื้อหาเกี่ยวกับอะไร
- เว็บเพจที่ไม่มีคำสำคัญ เพราะไม่สามารถนำเว็บเพจนั้นไปวิเคราะห์ข้อมูลได้

หลังจากทำการคัดเว็บเพจดังกล่าวออกแล้ว ทำให้เหลือเว็บเพจที่นำไปใช้วิเคราะห์ข้อมูลทั้งหมด 25,906 เว็บเพจ ซึ่งยังมีเว็บเพจส่วนที่ซ้ำในพนักงานและ วันที่ที่แตกต่างกันอยู่ เพราะฉะนั้นจะมีเว็บเพจที่ไม่ซ้ำกันจริงๆ 18,981 เว็บเพจ

4.4 ผลลัพธ์จากการหาค่าน้ำหนักให้กับคำสำคัญ

การหาค่าน้ำหนักให้คำสำคัญ หาเพียงแค่คำสำคัญที่อยู่ในแท็ก head เพราะว่าคำสำคัญที่ไม่อยู่ในแท็ก head จะไม่ถูกนำไปวิเคราะห์ เนื่องจากมีความสำคัญน้อยกว่าคำสำคัญที่อยู่ในแท็ก head โดยการวนรอบแต่ละคำสำคัญในทุกเว็บเพจว่ามีคำสำคัญนั้นอยู่หรือไม่ ถ้ามีให้บวกความถี่ IDF ไป 1 เท่ากับ 1 คำสำคัญต้องวนรอบ 18,981 ครั้ง แต่ถ้าคำสำคัญไหนที่เคยหาความถี่ IDF แล้วให้เก็บไว้ใน document corpus ถ้าเจอคำสำคัญนั้นในเอกสารอื่นก็ให้นำค่าความถี่ IDF นั้นมาใช้ได้เลย ดังรูปที่ 4.7

```
{
  "count": 744,
  "keyword": {
    "technological": {
      "tf": 4,
      "idf": 3,
      "weight": 1.9968317737981607,
      "in_head": "Yes"
    },
    "trend": {
      "tf": 6,
      "idf": 138,
      "weight": 1.1504265837727459,
      "in_head": "Yes"
    },
    "big": {
      "tf": 37,
      "idf": 217,
      "weight": 1.4256688937867892,
      "in_head": "Yes"
    },
    "data": {
      "tf": 42,
      "idf": 1829,
      "weight": 0.7781566546075298,
      "in_head": "Yes"
    },
    "analytics": {
      "tf": 10,
      "idf": 125,
      "weight": 1.228768403665805,
      "in_head": "Yes"
    },
```


รูป 4.7 ผลลัพธ์จากการหาค่าน้ำหนักด้วยวิธี TF-IDF บางส่วน

4.5 ผลลัพธ์จากการจำแนกประเภทของเว็บเพจ

เป้าหมายของประเภทเว็บเพจที่ต้องการจำแนกมี 2 ประเภทคือ ความรู้ และ ทั่วไป จากข้อมูลเว็บเพจที่ไม่ซ้ำทั้งหมด 18,981 เว็บเพจ เมื่อทำการสำรวจข้อมูลพบว่ามีเว็บเพจบางส่วนที่สามารถจำแนกประเภทได้เลยจาก URL และ โดเมน ได้แก่

- เว็บเพจที่เกี่ยวกับขายสินค้าและบริการออนไลน์ ให้จำแนกเป็นประเภท ทั่วไป โดยมีคำสำคัญดังกล่าวอยู่ใน URL ได้แก่ watson , shop.adidas , rcthai , kaidee , chilindo , getapple , weloveshopping , thaiticketmajor , lazada , ctrueshop , uniqlo , ygeshop , soapstation , central.co.th , cutepress , checkraka , th.priceprice , onitsukatiger , renthub , paeoni , playhouseth , healthy24hrs , promotiontoyou , dungdong , rakaball

- เว็บเพจที่เกี่ยวกับเรื่องทั่วไป ข่าว ให้จำแนกเป็นประเภท ทั่วไป โดยมีคำสำคัญดังกล่าวอยู่ใน URL ได้แก่ internship.mfu , soccersuck , jeban , ryoiireview , cheerball , music.sanook , truemoveh.truecorp , ais.co.th , dtac.co.th , bk-review , catdumb , sdbuffet , silamaneeresort , phonehip , loveyouna.online , jetstar , prakardproperty , nokscoot , lionairthai , maguro.co , xn--42cfal7c0d4a1d7a3d8ji.net , singaporefanclub , siamcomic , baanlaesuan , guchill , koithai , siamhahe , sdcdentist , ssru , khaisod , fpsthailand , show-anime , carrecent , mju

- เว็บเพจที่เกี่ยวกับข่าวเทคโนโลยี ความรู้ด้านไอที ให้จำแนกเป็นประเภท ความรู้ โดยมีคำสำคัญดังกล่าวอยู่ใน URL ได้แก่ fontawesome , mysql , quartz , forminit , somkiat.cc , oracle , python , bigdata , tensorflow , jsfiddle

รวมเว็บเพจที่ได้จำแนกประเภททั้งหมด 4,109 เว็บเพจซึ่งจะนำเว็บเพจจำนวน 4,000 เว็บเพจมาทำการสร้างโมเดลทำนายประเภทของเว็บเพจ ส่วนอีก 109 เว็บเพจจะทำมาทดสอบโมเดลโดยใช้ไลบรารี scikit-learn และ คุณลักษณะ (Feature) ที่นำมาใช้คือคำสำคัญ และค่าน้ำหนักในคำสำคัญใน คลังข้อมูลคำศัพท์ จำนวนคำสำคัญทั้งหมดที่นำมาเป็น คุณลักษณะ คือ 29,967 คำ ดังรูปที่ 4.8

	blackwidowchromav2	เบาะโฟม	communic	oraclegoldengate	ululicons222222	darkroom	mydhl	ตวนสุขนวิท	ahrefspace	จึง	...	รับควบคุม	alam	cspalignment
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
1	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
2	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
3	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
4	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

5 rows × 29967 columns

รูป 4.8 ผลลัพธ์ คุณลักษณะ ทั้งหมดบางส่วน

จากนั้นใช้ฟังก์ชัน SelectKBest ในการคัดเลือก คุณลักษณะ ที่จำเป็นกับโมเดลโดยตั้งไว้ที่ 1000 คุณลักษณะ โดยจะได้ผลลัพธ์ ดังรูปที่ 4.9

	website	orches	joomla	history	enterprisejob	translate	ceridian	current	tool	เกม	...	ซื้อ	cs	jsonpath	บริการ	dialog	register	override
0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0
1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.015342	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0
3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0
4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0

5 rows × 1000 columns

รูป 4.9 ผลลัพธ์ 1000 คุณลักษณะ บางส่วน

จากนำข้อมูลทั้ง 4000 เว็บเพจที่ทำการปรับ คุณลักษณะ เรียบร้อยแล้วนำไปสร้างโมเดลทำนายด้วยอัลกอริทึม นาอีฟเบย์ (Naive Bayes) เปรียบเทียบกับ แรนดอมฟอเรส (Random Forest) และซัพพอร์ตเวกเตอร์แมชชีน (Support Vector Machine) และทดสอบประสิทธิภาพของโมเดลด้วยวิธี 10 fold cross validation และทดสอบข้อมูลด้วย ข้อมูล 109 เว็บเพจที่เหลือ

ตาราง 4.1 การเปรียบเทียบโมเดล

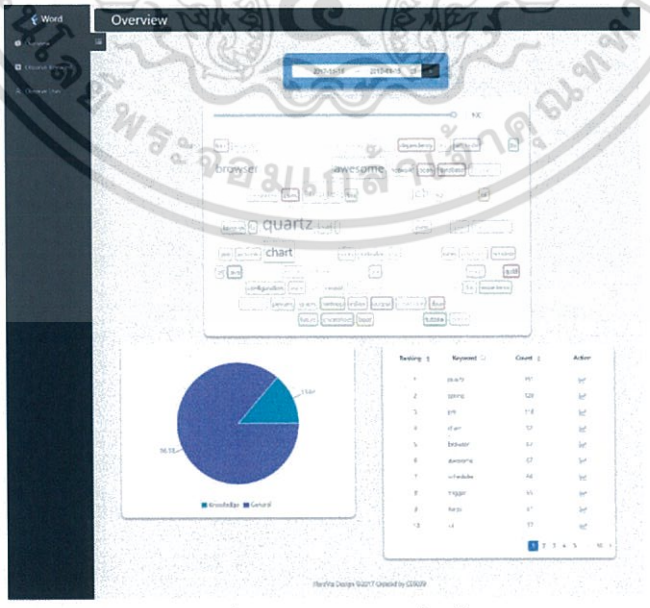
Model	Cross Validation Score	Test Accuracy
Naive Bayes	91.48%	97.39%
Random Forest	91.23%	94.77%
Support Vector Machine	90.04%	96.73%

จากตารางที่ 4.1 เลือกโมเดล Naive Bayes นำไปใช้จริงเพราะให้ผลได้ดีที่สุด

4.6 ผลลัพธ์จากการวิเคราะห์ข้อมูล

ข้อมูลที่วิเคราะห์จะถูกนำไปแสดงผลบนหน้าเว็บไซต์โดยแบ่งออกเป็น 3 ส่วนดังนี้

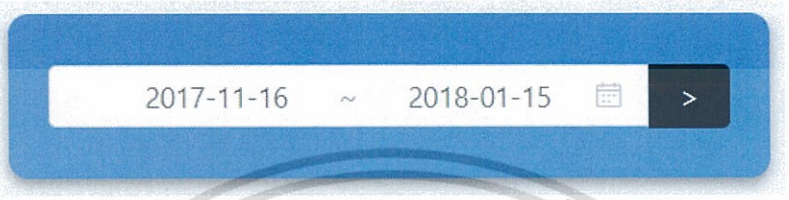
4.6.1 หน้า Overview



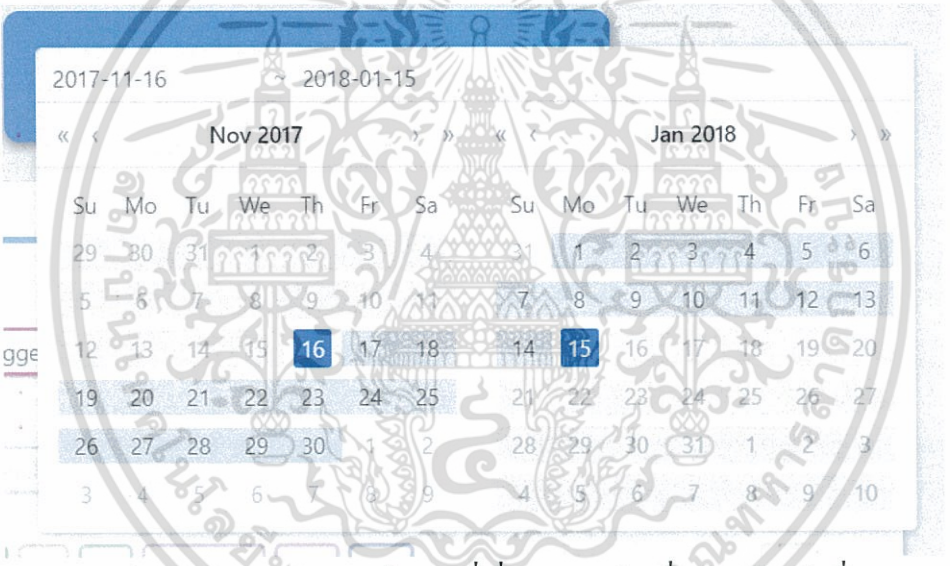
รูป 4.10 ผลลัพธ์จากการวิเคราะห์หน้า Overview

ในหน้า Overview เป็นหน้าสำหรับดูข้อมูลภาพรวมของคำสำคัญ ซึ่งได้แบ่งออกเป็น 4 ส่วนด้วยกัน คือ

1) **ตัวเลือกวันที่** ไว้สำหรับเลือกช่วงของวันที่ ที่ต้องการดูคำสำคัญได้ โดยวันที่จะถูกตั้งไว้วันที่ล่าสุดของข้อมูลที่มีในระบบดังรูปที่ 4.11 เมื่อไม่ได้ทำการเลือก และข้อมูลทั้งหมดจะถูกเปลี่ยนไปตามวันที่ ที่ถูกเลือกดังรูปที่ 4.12

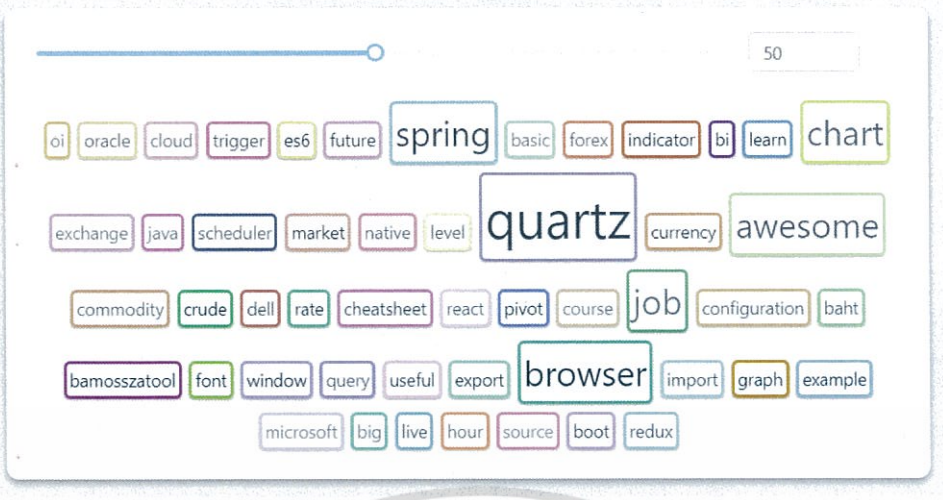


รูป 4.11 ลักษณะของตัวเลือกวันที่



รูป 4.12 ลักษณะของตัวเลือกวันที่เมื่อทำการคลิกเพื่อเลือกช่วงวันที่

2) **คำสำคัญยอดนิยม** บอกว่าคำสำคัญนั้นถูกสนใจโดยพนักงานมากที่สุดในช่วงเวลาที่ถูกคัดกรองมากแค่ไหน โดยกล่องที่มีขนาดใหญ่คือ มีการเข้าถึงเยอะกว่ากล่องขนาดเล็ก โดยสามารถปรับจำนวนคำสำคัญที่แสดงได้ถึง 100 คำสำคัญ โดยมีค่าตั้งต้นที่ 50 คำ ดังรูปที่ 4.13



รูป 4.13 คำสำคัญยอดนิยมที่ถูกตั้งไว้ที่ 50 คำ

3) **กราฟแผนภูมิวงกลม** บอกสัดส่วนของเว็บเพจในระบบทั้งสองประเภท คือ ความรู้ และทั่วไป โดยบอกเป็นเปอร์เซ็นต์ ดังรูปที่ 4.14

รูป 4.14 กราฟแผนภูมิวงกลมแสดงสัดส่วนเว็บเพจความรู้ และทั่วไป

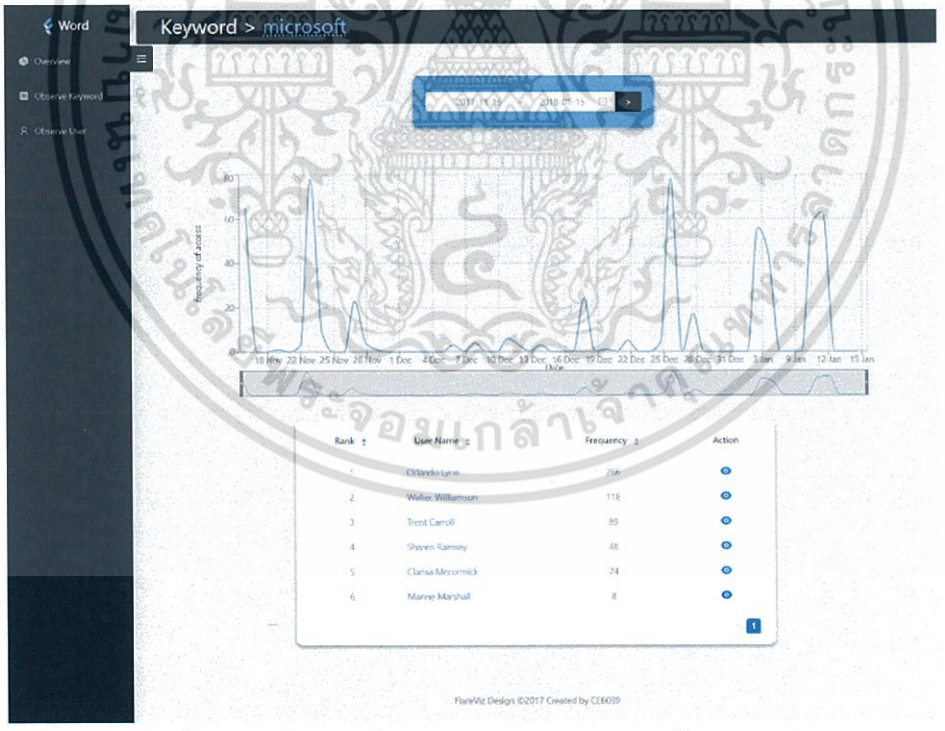
4) **ตารางแจกแจงความถี่คำสำคัญ** เป็นตารางไว้สำหรับดูคำสำคัญทั้งหมดในแต่ละช่วงวันที่ ที่ได้ทำการเลือก สามารถทำการจัดอันดับคำสำคัญ และค้นหาคำสำคัญที่สนใจได้ เมื่อได้คำที่สำคัญที่สนใจแล้วสามารถเลือก ปุ่ม ไอคอนกราฟ เพื่อไปยังหน้า Observe Keyword เพื่อดู ข้อมูลเชิงลึกของข้อมูลคำสำคัญได้ ดังรูปที่ 4.15

Ranking	Keyword	Count	Action
1	quartz	191	
2	spring	128	
3	job	117	
4	chart	115	
5	browser	100	
6	awesome	80	
7	forex	78	
8	oi	68	
9	react	67	
10	microsoft	67	

1 2 3 4 5 ··· 10 >

รูป 4.15 ตารางแจกแจงความถี่ของคำสำคัญ

4.6.2 หน้า Observe Keyword



รูป 4.16 ผลลัพธ์จากการวิเคราะห์หน้า Observe Keyword ด้วยคำสำคัญ Microsoft

จากรูปที่ 4.16 หน้า Observe Keyword เป็นหน้าสำหรับดูข้อมูลเชิงลึกของคำสำคัญว่าถูกเข้าถึงมากแค่ไหน และพนักงานคนใดบ้างที่เข้าถึงคำสำคัญนี้ ประกอบไปด้วย 4 ส่วนหลักๆ ด้วยกัน คือ

1) **แถบแสดงผลการเลือกคำสำคัญ** จะเป็นคำที่แสดงอยู่บนแถบด้านบนหลังจากที่ได้เลือกคำสำคัญ ที่สนใจจากหน้า Overview แล้ว เพื่อบอกผู้ใช้ว่า ตอนนี้ได้ทำการเลือกคำสำคัญนี้อยู่ ดังรูปที่ 4.17

รูป 4.17 แถบแสดงผลมีคำสำคัญคือ Microsoft อยู่

2) **ตัวเลือกวันที่** สามารถเลือกช่วงของวันที่ ที่ต้องการดูได้ ตัวอย่างเหมือน รูปที่ 4.11 และ 4.12

3) **กราฟเส้นแสดงความถี่ของคำสำคัญ** ใช้แสดงค่าความถี่ของการเลือกคำสำคัญนั่นอยู่ ซึ่งตัวอย่างนี้คือคำว่า microsoft ที่ในแต่ละวันจะมีพนักงานค้นคำสำคัญนี้ ดังรูปที่ 4.18

รูป 4.18 กราฟเส้นแสดงความถี่ของคำว่า Microsoft

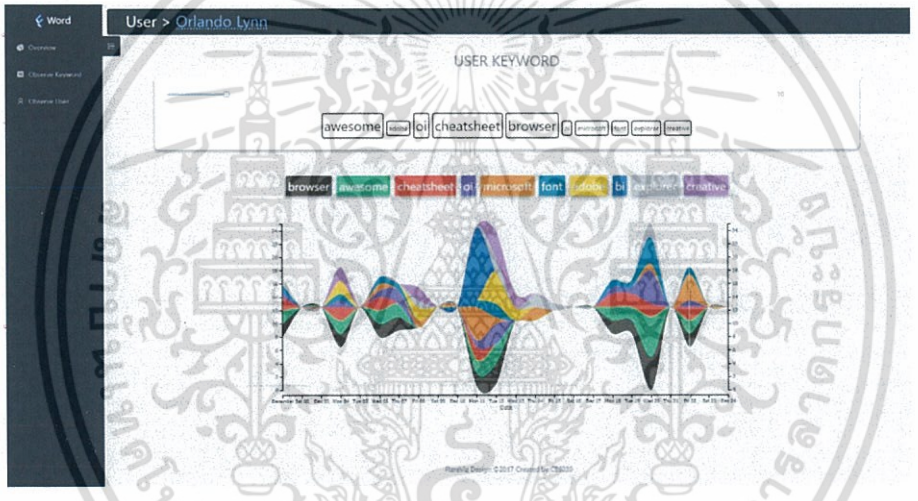
4) **ตารางชื่อของพนักงานที่ เข้าถึง คำสำคัญ ที่กำลังเลือกอยู่** สามารถที่จะเลือกชื่อพนักงานที่สนใจแล้วจะเข้าไปสู่หน้า Observe User เพื่อดูพฤติกรรมการเข้าถึงคำสำคัญอื่นๆ ได้ หรือจะเข้าผ่านปุ่ม ไอคอนรูปตา ก็จะทำการลิงค์เข้าไปสู่ หน้า Observe User ได้เช่นกัน ดังรูปที่ 4.19

Rank	User Name	Frequency	Action
1	Orlando Lynn	266	
2	Walter Williamson	118	
3	Trent Carroll	89	
4	Sharen Ramsey	48	
5	Clarisa Mccormick	24	
6	Marine Marshall	8	

1

รูป 4.19 ตารางแจกแจงความถี่พนักงานที่เข้าถึงคำสำคัญนั้น

4.6.3 หน้า Observe User



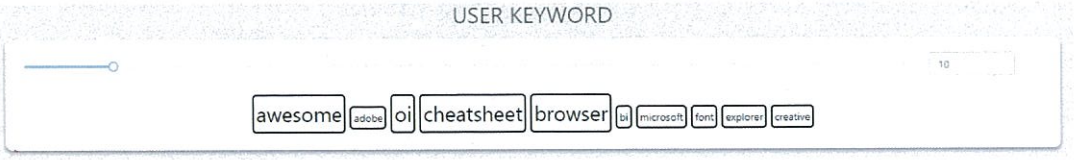
รูป 4.20 ผลลัพธ์จากการวิเคราะห์หน้า Observe User ด้วย User ชื่อ Orlando Lynn

ดังรูปที่ 4.20 สำหรับหน้า Observe User เป็นหน้าสำหรับดูข้อมูลเชิงลึกของพนักงานที่สนใจว่ามีพฤติกรรมในการเข้าถึงคำสำคัญใดบ้างและคำยอดนิยมนั่นมีการติดตามหรือมีความสนใจอยู่หรือไม่ ประกอบด้วย 3 ส่วนหลักๆ คือ

1) **แถบแสดงผลการเลือกพนักงาน** จะเป็นชื่อที่ถูกเข้ารหัสโดยจะแสดงอยู่บนแถบด้านบนหลังจากที่ได้เลือกชื่อเข้ามาแล้ว จากหน้า Observe Keyword เพื่อบอกผู้ใช้ว่า ตอนนี้ได้ทำการเลือกพนักงานคนนี้อยู่ ดังรูปที่ 4.21

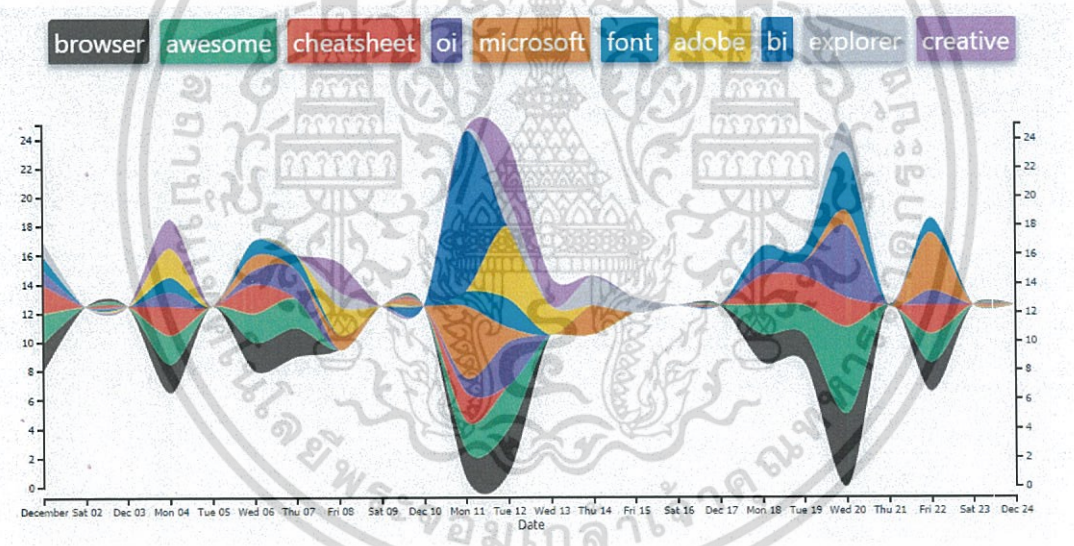
รูป 4.21 แถบแสดงผลมีชื่อของพนักงานที่เลือกไว้ คือ Orlando Lynn อยู่

2) **แผนภาพคำสำคัญที่พนักงานเข้าถึงมากที่สุด** โดยจะเป็นคำสำคัญ 10 คำ ในตอนต้นสามารถที่เลื่อนดูคำได้มากสุด 100 คำ ดังรูปที่ 4.22



รูป 4.22 แผนภาพคำสำคัญที่พนักงานเข้าถึงมากที่สุด

3) **กราฟ สตรีม (Stream)** เป็นกราฟที่แสดงความต่อเนื่องในการเข้าถึงคำสำคัญในแต่ละวันโดย 1 สีคือ 1 คำสำคัญถ้าสีใดมีความกว้างของกราฟมากแปลว่าคำสำคัญนั้นมีการเข้าถึงมาก โดยแบ่งเป็น 10 สี คือ คำสำคัญที่พนักงานเข้าถึงมากที่สุด 10 คำสำคัญ ดังรูปที่ 4.23



รูป 4.23 กราฟสตรีม

4.7 สรุปผลการทดสอบ

4.7.1 การหาคำสำคัญและค่าน้ำหนัก TF-IDF

ในการตัดคำจากเว็บเพจด้วยวิธีนี้สามารถให้ผลลัพธ์คำสำคัญที่มีความแม่นยำที่พอรับได้ในเว็บเพจส่วนใหญ่ที่นำมาทดสอบเฉพาะภาษาอังกฤษ โดยอัลกอริทึม เวิร์ดเลมาไทเซอร์ (word lemmatizer) มีความแม่นยำในการหารากศัพท์มากกว่าอัลกอริทึม เวิร์ดสเตมมิ่ง (word stemming) ที่เหลือเนื่องจาก เวิร์ดเลมาไทเซอร์ มี เวิร์ดเน็ต ที่เป็นฐานข้อมูล ของคำ ทำให้มีการระวังเรื่องความหมายของคำหลังจากตัดคำลงท้ายออกไปทำให้การใช้อัลกอริทึม เวิร์ดเลมาไทเซอร์ ให้

ผลลัพธ์ค่า TF-IDF แม่นยำมากกว่า เวิร์ดสเตมมิ่ง แต่ยังมีปัญหาใน และมีการแปลงรูปจากพหูพจน์เป็นเอกพจน์ และเรื่องของคำที่มีความคล้ายกัน (synonyms) เช่น ocean, sea หรือ dog, canine

นอกจากนี้มีปัญหาในเรื่องของความหลากหลายของคำใน เว็บเพจ เช่น เว็บเพจ ภาษา C# ในการตัดคำ คำว่า C# จะถูกตัด # ออกไปเพราะถูกมองว่าเป็นอักขระพิเศษ ทำให้คำสำคัญจากเว็บเพจนี้ผิดเพี้ยนไป

ส่วนภาษาไทยยังมีความผิดพลาดในการตัดคำ และเว็บเพจส่วนใหญ่เป็นภาษาอังกฤษ และคำสำคัญในภาษาไทยปรากฏบนแท็ก head เป็นจำนวนน้อยทำให้ส่วนใหญ่ไม่ถูกนำมาพิจารณาเป็นคำสำคัญ

4.7.2 การจัดประเภทคำสำคัญ

นำโมเดลที่ได้ไปทำนายประเภทของเว็บเพจที่เหลือทั้งหมดได้สัดส่วนจำนวนของประเภทเว็บเพจเป็น ความรู้ 14.59% และ ทั่วไป 85.41% ซึ่งจากการสำรวจผลลัพธ์ เว็บเพจที่อยู่ในประเภท ทั่วไป ทำนายถูกน้อยกว่า เว็บเพจที่อยู่ในประเภท ความรู้ ที่ทายได้แม่นยำมากกว่า

4.7.3 การวิเคราะห์ข้อมูล

- หน้า overview ผลลัพธ์จากหน้า overview ส่วนที่เป็นคำสำคัญยอดนิยมพบว่ามีคำสำคัญในส่วนของ quartz , spring , react , microsoft เป็นส่วนใหญ่แสดงว่าพนักงานมีความสนใจในเรื่อง Software Development ในภาษา java และ javascript มากในช่วงเวลาดังกล่าว และรองลงมามีบางส่วนที่สนใจในเรื่อง forex ก็คือตลาดแลกเปลี่ยนเงิน (trade)
- หน้า observe keyword หน้าเพจนี้ต้องเลือกคำสำคัญมาจากหน้า overview ซึ่งจะยกตัวอย่างผลลัพธ์จากคำสำคัญ microsoft พบว่ามีความถี่ในการเข้าถึงสูงในหลายวัน จากพนักงาน 6 คน ซึ่งพนักงานที่ชื่อ Orlando Lynn มีการเข้าถึงรวมมากที่สุด
- หน้า observe user หน้าเพจนี้ต้องเลือก user มาจากหน้า observe keyword ซึ่งจะยกตัวอย่างผลลัพธ์จากพนักงานที่ชื่อว่า Orlando Lynn พบว่าคำสำคัญที่มีความถี่ในการเข้าถึงสูงของพนักงานคนนี้คือ browser , awesome , cheatsheet , oi , microsoft , font , adobe , bi , explorer , creative จึงสามารถคาดเดาได้ว่าพนักงานคนนี้สนใจเกี่ยวกับงานทางด้านออกแบบ หรือศิลปะ

4.8 อภิปรายผลการทำวิจัย

จากการเริ่มต้นพัฒนาโปรเจ็กต์พบว่า เกิดปัญหาตลอดการดำเนินงานตั้งแต่เริ่มสร้างโมเดลที่ใช้ในการหาคำสำคัญในภาษาอังกฤษในภาคการศึกษาที่ 1 จน ถึงตอนสร้าง API ให้ฝั่งหน้าเว็บเรียก

ข้อมูลที่วิเคราะห์ไปแสดงผลแต่สามารถแก้ปัญหาได้จนได้ผลลัพธ์ที่ดีพอใช้ โดยปัญหาที่เกิดขึ้น มีสี่ส่วนหลักๆ ดังนี้

4.8.1 ผลลัพธ์คำสำคัญ

ในการพัฒนาโมเดลที่ใช้ในการหาคำสำคัญภาษาอังกฤษในภาคการศึกษาที่ 1 ในช่วงที่เริ่มพัฒนาพบปัญหาเรื่องผลลัพธ์จากการตัดคำมากมายเพราะไลบรารีที่เลือกใช้ให้ผลลัพธ์ต่างกัน แต่เลือกใช้ไลบรารี NLTK และภายในไลบรารีเดียวกันก็มีอัลกอริทึมที่แตกต่างกัน ซึ่งตอนแรกใช้เพียงเวิร์ดเลมมาไทเซชัน (word lemmatization) เพราะให้ผลดีกว่า แต่ก็ยังติดปัญหาเรื่องคำคุณศัพท์ (Adjective) ที่เวิร์ดเลมมาไทเซชันไม่ตัดคำให้ ทำให้ต้องใช้เวิร์ดสเตมมิ่งในการตัดคำคุณศัพท์ จึงต้องใช้การผสมกันระหว่างอัลกอริทึมสองตัว ซึ่งให้ผลการตัดคำที่ดี

จากนั้นติดปัญหาเรื่องค่าน้ำหนักที่ใช้คือ TF-IDF โดยค่า TF นั้นคือค่าความถี่ของคำในเว็บเพจนั้น สูตรที่ใช้คือ ค่าความถี่ของคำนั้นหารด้วยจำนวนคำทั้งหมดในเว็บเพจ เกิดเหตุการณ์ที่ว่าบางคำในเว็บเพจมีความถี่ที่ต่างกันมากทำให้ค่า TF มีความแตกต่างกันเกินไป และแต่ละเว็บเพจมีจำนวนคำที่แตกต่างกันมากทำให้ส่งผลต่อค่า TF ทำให้ผลลัพธ์ค่าน้ำหนักของคำสำคัญผิดพลาด แต่เนื่องจากในภาคการศึกษาที่ 1 ใช้ข้อมูลเว็บเพจที่สร้างขึ้นมาเอง จงใจให้ข้อมูลมีทิศทางที่ต้องการทำให้เว็บเพจไม่มีความหลากหลาย จึงไม่เห็นข้อผิดพลาดของค่าน้ำหนักที่ใช้ แต่เมื่อใช้วิธีในข้อมูลจริงจากองค์กรในภาคการศึกษาที่ 2 จึงเห็นว่าวิธีแบบเดิมทำให้ค่าน้ำหนักผิดพลาด จึงต้องเปลี่ยนมาใช้สูตรการหาค่า TF ใหม่คือ ดับเบิ้ลนอร์มัลไลเซชัน (double normalization) และข้อมูลเว็บเพจก็หลากหลายมากขึ้นทำให้ผลลัพธ์ของค่าน้ำหนักดูเป็นจริงมากขึ้น

4.8.2 การทำความสะอาดข้อมูล

ข้อมูลที่ได้จากองค์กรมีขนาดใหญ่ถึง 6 GB แต่ส่วนใหญ่ล้วนเป็นข้อมูลที่ใช้ไม่ได้ เนื่องจากการจะนำเว็บเพจแต่ละเว็บเพจมาวิเคราะห์ได้ ต้องสามารถดึงเนื้อหาข้างในออกมาได้ ซึ่งบางส่วนก็จะมีคอลัมน์ที่บอกว่าเว็บเพจนั้นเป็นชนิดไหน ซึ่งสามารถคัดกรองออกได้ในระดับหนึ่ง และนำเว็บเพจที่เหลือนำไปหาคำสำคัญและเก็บลงดาต้าเบส ทุกเว็บเพจต้องผ่านขั้นตอนการ get request เพื่อเอาเนื้อหานั้นออกมาตัดคำ แต่มีหลายเว็บเพจใช้เวลานานกว่าปกติในการประมวลผล และสุดท้ายก็ไม่สามารถเอาเนื้อหานั้นออกมาได้ ทำให้ขั้นตอนนี้ล่าช้าไปเพราะต้องทำการสำรวจข้อมูลประเภทนั้นและคัดออกไป เพราะไม่ทราบว่าจะมีเว็บเพจแบบไหนที่เกิดข้อผิดพลาดบ้าง

ข้อมูลที่ได้มามีคอลัมน์ที่บอกว่าเว็บเพจอยู่ในประเภทอะไร แต่มีความผิดพลาดมาก ตรงที่เว็บเพจเดียวกันแต่เป็นคนละประเภทกันทำให้ต้องตัดคอลัมน์นั้นออกไป และทำการจัดประเภทของเว็บเพจเองด้วยแมชชีนเลิร์นนิ่งซึ่งก็ให้ผลได้ดีพอใช้

4.8.3 คอมพิวเตอร์ที่ใช้ในการประมวลผล

เนื่องจากข้อมูลที่ได้จากองค์กรเป็นข้อมูลขนาดใหญ่ คอมพิวเตอร์ที่ใช้ในการประมวลผลมีเพียงเครื่องเดียวทำให้เห็นผลลัพธ์จากการตัดคำ ค่าน้ำหนัก ทำความสะอาดข้อมูลแต่ละครั้งล่าช้า เนื่องจากภาษา Python คอมไพล์ได้ช้ากว่าภาษาอื่นๆ ส่วนใหญ่ จึงต้องแก้ไขโดยการเขียนโปรแกรมให้ประมวลผลแบบขนาน (Parallel Computing) ใช้ตัวประมวลผลบนซีพียูทั้งหมด และใช้ระบบแคช (cache) ในข้อมูลที่เคยประมวลผลไปแล้วให้เก็บไว้บนหน่วยความจำ และสร้างฐานข้อมูลสำหรับเก็บผลลัพธ์การประมวลผลช่วยเผื่อว่าหน่วยความจำเต็ม ซึ่งทำให้ไม่ต้องประมวลผลซ้ำ ในทุกกระบวนการแม้แต่ผลลัพธ์การวิเคราะห์ที่ต้องส่งให้หน้าเว็บไปแสดงผล เนื่องจากการเรียกดู 1 ครั้งต้องคำนวณผลลัพธ์จากทุกเว็บเพจ ซึ่งมีเว็บเพจเป็นจำนวนมากและแต่ละเว็บเพจก็มีคำสำคัญเป็นจำนวนมาก หลังจากได้ใช้วิธีดังกล่าวทำให้เห็นผลลัพธ์เร็วขึ้น และเวลาในการพัฒนาสั้นลงมาก

4.8.4 ผลลัพธ์การวิเคราะห์ข้อมูล

เมื่อทำการคำนวณผลลัพธ์คำสำคัญแล้ว นำไปแสดงบนหน้าเว็บพบว่า หัวข้อส่วนใหญ่ที่พนักงานส่วนใหญ่เป็นเรื่องทั่วไป และคำสำคัญที่นำมาแสดงผลนั้นสามารถแสดงผลได้จำนวนจำกัด ทำให้คำสำคัญของหัวข้อที่มีประโยชน์นั้นถูกบดบัง แก้ไขโดยการคัดคำสำคัญออกมาคำนวณเฉพาะประเภท ความรู้ ทำให้ภาพรวมคำสำคัญดูดีขึ้น

ผลลัพธ์ของวิจัยสามารถนำไปพัฒนาต่อได้ในเรื่องของการหาคำสำคัญในภาษาไทยที่ยังมีความผิดพลาดสูงอยู่ แต่โดยรวมเว็บเพจที่เป็นภาษาอังกฤษสามารถหาคำสำคัญได้ดี และเมื่อแยกแสดงผลเฉพาะหัวข้อที่มีประโยชน์ต่อองค์กรแล้วสามารถบอกได้ว่าพนักงานสนใจในหัวข้อหรือประเด็นใดบ้าง

บทที่ 5
บทสรุปและข้อเสนอแนะ

5.1 บทสรุป

จากการพัฒนาโปรเจกต์ ข้อมูลการใช้งานอินเทอร์เน็ตเริ่มต้นมีข้อมูล 13 คอลัมน์ 300,000,000 แถวใช้ได้จริง 6 คอลัมน์ 7,500,000 แถว เพราะว่าข้อมูลส่วนใหญ่ถูกคัดกรองออกเนื่องจากมีเลขสถานะการตอบรับที่ไม่ใช่ 200 (HTTP Status Code 200), เนื้อหาบนเว็บเพจไม่ใช่ text/html, http method เป็น POST และส่วนที่เป็นเว็บเพจเครือข่ายสังคม (Social Network) ที่ติดปัญหาเรื่องความเป็นส่วนตัวในเนื้อหาของบุคคล

หลังจากที่ทำความสะอาดข้อมูลเบื้องต้นแล้ว ต่อมาทำการตัดคำบนเว็บเพจและหาคำสำคัญของข้อมูลเว็บเพจพนักงาน 100 คน พบว่ามีข้อความบนเว็บเพจเป็นจำนวนมากจนเกินไปจนต้องใช้เนื้อหาเพียงแค่ส่วนที่อยู่บนแท็ก head คือ title, h1, h2, h3, h4, h5, h6 และโมเดลที่ใช้ในการตัดคำได้พัฒนามาจากภาคการศึกษาที่ 1 คือเพิ่มส่วนที่ตัดคำภาษาไทยเข้าไปพบว่าในส่วนของภาษาไทยยังมีความผิดพลาด และปรับปรุงอัลกอริทึมที่ใช้หาค่าน้ำหนัก TF-IDF ทำให้คำสำคัญของเว็บเพจดูเป็นจริงมากขึ้น จากนั้นนำข้อมูลเก็บลงฐานข้อมูลและนับความถี่ของเว็บเพจที่ซ้ำ มี 38,402 เว็บเพจ

หลังจากที่ทำการสำรวจข้อมูลแล้วก็ยังมีเว็บเพจที่ใช้ไม่ได้อีกเนื่องจากเป็นเว็บเพจที่ไม่ให้ประโยชน์ต่อการวิเคราะห์ และเป็นเว็บเพจที่พนักงานไม่ได้เข้าถึงด้วยความสนใจเช่นเว็บลงชื่อเข้าใช้อินเทอร์เน็ตขององค์กร เมื่อคัดเว็บเพจดังกล่าวออกแล้ว จะเหลือ 25,906 เว็บเพจ

ขั้นตอนการหาค่า IDF จะใช้เวลานานเพราะต้องวนรอบทุกเว็บเพจทุกคำสำคัญ ยกเว้นคำที่เคยคำนวณแล้วจะใช้ค่าจากคลังข้อมูลมาเก็บได้เลย ทำให้การพัฒนาโปรเจ็กต์ล่าช้าไป จึงใช้เวลาช่วงนี้ในการพัฒนาโมเดลในการจัดประเภทของเว็บเพจโดยใช้แมชชีนเลิร์นนิ่ง ทำการจัดประเภทข้อมูลด้วยตัวเอง 4,109 เว็บเพจนำมาสร้างโมเดล 4,000 เว็บเพจ อีก 109 เว็บเพจใช้วัดประสิทธิภาพ จากนั้นทำการเปรียบเทียบอัลกอริทึมแมชชีนเลิร์นนิ่งแล้ว Naive Bayes ให้ประสิทธิภาพของโมเดลดีที่สุด และนำไปใช้จัดประเภทเว็บเพจที่เหลือและสำรวจดูผลลัพธ์พบว่า มีความถูกต้องพอใช้ ได้สัดส่วนจำนวนประเภทของเว็บเพจเป็นความรู้ (Knowledge) 14.59 เปอร์เซ็นต์ และเรื่องทั่วไป (General) 85.41 เปอร์เซ็นต์

ขั้นตอนต่อไปคือการพัฒนา API เพื่อนำไปแสดงผลบนเว็บ ผลลัพธ์ครั้งแรกมีคำสำคัญส่วนที่เป็นเรื่องทั่วไปเยอะมาก ทำให้ต้องเลือกแสดงผลเฉพาะคำสำคัญจากเว็บเพจประเภทความรู้แต่การเรียกดูแต่ละครั้งใช้เวลานานมากกว่า 5 วินาที โดยเฉพาะหน้าภาพรวม (overview) ทำให้ต้องเก็บผลลัพธ์ของแต่ละช่วงเวลาทั้งหมดเก็บไว้ในฐานข้อมูลในตารางใหม่ ทำให้การเรียกดูแต่ละครั้งเร็วขึ้นแทบจะทันที แต่หน้ารายละเอียดคำสำคัญไม่สามารถทำแบบนั้นได้เพราะว่าไม่สามารถเก็บ

ข้อมูลของทุกคำทั้งหมดเนื่องจากมีจำนวนคำมากจนเกินไปและเมื่อคูณกับจำนวนช่วงเวลาที่เป็นไปได้ทั้งหมด ทำให้ไม่คุ้มกับเวลาที่ต้องคำนวณผลลัพธ์เหล่านั้นและเก็บลงฐานข้อมูล และหน้ารายละเอียดพนักงานไม่จำเป็นต้องเก็บผลลัพธ์ที่เคยเรียกหน้าเว็บเพจนั้นไว้ เพราะเวลาที่ใช้ในการคำนวณใช้เวลาไม่มากอยู่แล้ว

5.2 ปัญหาอุปสรรคและแนวทางแก้ไข

คำที่พ้องความหมาย และ คำที่มีความหมายคล้ายกันเป็นคนละคำสำคัญ โดยแนวทางแก้ไขคือ ทำการเพิ่มส่วนที่เป็น คำพ้องความหมาย (synonyms) และ คำที่มีความคล้ายกัน (word similarity) ประกอบเข้าไปในโมเดลเพื่อเพิ่มความแม่นยำของค่า TF-IDF

มีคำที่เป็นคำสำคัญที่มีค่าน้ำหนักสูง แต่ไม่ใช่คำสำคัญที่ถูกต้องปรากฏอยู่ในรายการ มีผลทำให้ผลลัพธ์ของการค้นและจัดประเภทของเว็บเพจผิดพลาดไป โดยแนวทางแก้ไขคือ ปรับปรุงคำในคำหยุด ให้ตรงประเด็นมากยิ่งขึ้นเพื่อลดคำที่ไม่ต้องการออกไป

คำสำคัญในภาษาไทยและภาษาอังกฤษที่มีความหมายเดียวกันถูกมองว่ามีเป็นคนละคำกัน โดยแนวทางแก้ไขคือ ทำระบบ word corpus ที่เก็บคำที่มีความหมายเหมือนกันของทั้งสองภาษา

ผลลัพธ์จากการจัดประเภทของเว็บเพจยังไม่แม่นยำมากพอ เนื่องจากคำมีความหลากหลายมากเกินไป โดยแนวทางแก้ไขคือ ใช้ข้อมูลมาสร้างโมเดลให้มากขึ้นและเลือกอัลกอริทึมที่เหมาะสมกับข้อมูลนั้น

คำสำคัญบางคำสำคัญที่เป็นวลีถูกตัดแยกออกจากกันและถูกคิดค่าน้ำหนักแยกกันทำให้ถูกมองว่าเป็นคนละคำกันและทำให้ความหมายผิดเพี้ยนไปเช่นคำว่า big data ถูกแยกออกจากกัน โดยแนวทางแก้ไขคือ ปรับปรุงอัลกอริทึมในการตัดคำให้สามารถตัดคำเป็นวลีได้และพัฒนาค่าน้ำหนักคำสำคัญใหม่

ไม่สามารถหาข้อมูลเชิงลึกของข้อมูลได้มากพอทำให้การแสดงผลข้อมูลไม่มีประสิทธิภาพ โดยแนวทางแก้ไขคือ ศึกษางานวิจัยอื่นๆ ที่เกี่ยวกับการวิเคราะห์ข้อมูลให้มากขึ้น

5.3 แนวทางการพัฒนาต่อ

สามารถปรับปรุงอัลกอริทึมที่ใช้ในการหาคีย์เวิร์ดของทั้งภาษาไทยและภาษาอังกฤษและจำแนกประเภทของเว็บเพจให้มีประสิทธิภาพมากขึ้นได้

เพิ่มโจทย์ที่ต้องการวิเคราะห์ข้อมูลให้มากขึ้นเพื่อดูข้อมูลเชิงลึกอื่นๆ ของข้อมูลให้ระบบมีความหลากหลายมากขึ้น เช่นการวิเคราะห์พฤติกรรมของพนักงานจากข้อมูลการใช้งานอินเทอร์เน็ต

บรรณานุกรม

Julie, S. and Noah , I. 2011. Designing Data Visualizations: Representing Informational Relationships. O'Reilly Media, Inc.

TextMiner. 2014. Dive Into NLTK, Part II: Sentence Tokenize and Word Tokenize [Online]. Available : http://textminingonline.com/dive-into-nltk-part-ii-sentence-tokenize-and-word-tokenize

Panyatham. 2538. การให้น้ำหนักคำ [Online].Available : https://th.wikipedia.org/wiki/การให้น้ำหนักคำ

Daniel Tunkelang. 2017. Stemming and Lemmatization. [online]. Available : https://queryunderstanding.com/stemming-and-lemmatization-6c086742fe45

Opencooper. 2017. WordNet. [online]. Available : https://en.wikipedia.org/wiki/WordNet

Motioninfostudio. 2013 .เป็นมากกว่าข้อมูลด้วย DATA VISUALIZATION [Online].Available : http://www.motioninfostudio.com/เป็นมากกว่าข้อมูลด้วย-data-visua/#.WhdtAubObtR

กิรดา กลีบมาลัย. 2558. Graph & Chart [Online].Available : http://v54-30037.blogspot.com/2015/01/graph-chart.html

BattyBot. 2017. Data visualization.[Online]. Available : https://en.wikipedia.org/wiki/Data_visualization

Sakul Montha. 2017. Redis คืออะไร.[Online]. Available: https://medium.com/@iamgique/what-redis-is-4381ff32880d