การค้นหาเว็บไซต์แบบเจาะจง

SPECIFIC WEB SEARCH

โดย

นันทวัฒน์ ไชยรัตน์

NANTAWAT CHAIRAT

รายงานนี้เป็นส่วนหนึ่งของวิชาการศึกษาอิสระ
หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
คณะเทคโนโลยีสารสนเทศ
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ภาคเรียนที่ 2 ปีการศึกษา 2552

SPECIFIC WEB SEARCH :

DESIGN AND IMPLEMENTATION

NANTAWAT CHAIRAT

A REPORT SUBMITTED IN PARTIAL FULFILLMENT OF THE

REQUIREMENTS OF THE COURSE

INDEPENDENT STUDY

MASTER OF SCIENCE PROGRAM IN INFORMATION TECHNOLOGY

FACULTY OF INFORMATION TECHNOLOGY

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2 / 2009

ใบรับรองการศึกษาอิสระ (Independent Study)

เรื่อง

การค้นหาเว็บไซต์แบบเจาะจง

SPECIFIC WEB SEARCH

นายนันทวัฒน์ ไชยรัตน์

รหัสประจำตัว 51066418

ขอรับรองว่ารายงานฉบับนี้ ข้าพเจ้าไม่ได้คัดลอกมาจากที่ใด

รายงานฉบับนี้ได้รับการตรวจสอบและอนุมัติให้เป็นส่วนหนึ่งของ

การศึกษาวิชาการศึกษาอิสระ หลักสูตรวิทยาศาสตรมหาบัณฑิต (เทคโนโลยีสารสนเทศ)

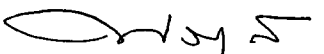
ภาคเรียนที่ 2 ปีการศึกษา 2552

.....อาจารย์ที่ปรึกษา

(รศ.ดร. วรพจน์ กรีสุระเดช)

.....กรรมการสอบ

(รศ.ดร. อาริต ธรรมโน)



.....กรรมการสอบ

(ผศ.ดร. พรฤดี เนติโสภากุล)

หัวข้อ	การค้นหาเว็บไซต์แบบเจาะจง
นักศึกษา	นายนันทวัฒน์ ไชยรัตน์
รหัสนักศึกษา	51066418
ปริญญา	วิทยาศาสตรมหาบัณฑิต
สาขาวิชา	เทคโนโลยีสารสนเทศ
แขนงวิชา	วิทยาการสารสนเทศ
ปีการศึกษา	2552
อาจารย์ที่ปรึกษา	รศ.ดร. วรพจน์ กรีสุระเดช

บทคัดย่อ

ปัจจุบันข้อมูลที่อยู่ในเว็บไซต์นั้นมีจำนวนมาก และมีแนวโน้มจะมากขึ้นเรื่อยๆ การที่มีข้อมูลมากมายขนาดนี้นั้นจำเป็นจะต้องมีเครื่องมือที่เข้ามาช่วยในการจัดการ เพื่อให้เกิดการใช้ประโยชน์ได้มากที่สุด ปัจจุบันเสิร์ชเอนจินเป็นเครื่องมือหนึ่งที่มีความสามารถและมีผู้ใช้งานจำนวนมาก หากมีการเพิ่มวิธีการในการทำงานของเสิร์ชเอนจิน เพื่อให้ได้ผลลัพธ์ในการทดลองที่ดีขึ้นเพียงเล็กน้อย แต่จะส่งผลกับข้อมูลจำนวนมหาศาลเมื่อใช้งานจริง ในบทความนี้จะเสนอแนวคิดในการเพิ่มประสิทธิภาพของเสิร์ชเอนจิน โดยใช้หลักการของอินโฟสไปเดอร์

Title	Specific Web Search.
Student	Mr.Nantawat Chairat
Student ID.	51066418
Degree	Master of Science
Program	Information Technology
Major	Information Science
Academic Year	2009
Advisor	Assoc.Prof. Dr.Worapoj Kreesuradej

ABSTRACT

Current information on the site are numerous and are increasingly likely. The size of this data is that many need to have tools to help manage. To use the most current Search Engine is a tool that is capable and has many users. If the increase in the working methods of the Search Engine to get the test results only slightly better. But will affect the masses of information on actual use. This article will offer ideas on how to optimize your Search Engine, using principles of Infospider.

กิตติกรรมประกาศ

โครงการพัฒนาระบบงานนี้ ประสบความสำเร็จได้ด้วยความช่วยเหลือและการสนับสนุนจากบุคคลหลายท่าน ผู้เขียนใคร่ขอแสดงความระลึกถึงบุคคลสำคัญผู้อยู่เบื้องหลัง ดังต่อไปนี้

คุณพ่อและคุณแม่สำหรับกำลังใจและทุกสิ่งทุกอย่างจนผู้เขียนสามารถมีวันนี้ได้

รศ.ดร. วรพจน์ กรีสุระเดช เป็นอาจารย์ที่ปรึกษาโครงการและอาจารย์ที่ปรึกษาสัมมนาที่กรุณาให้คำปรึกษา ให้กำลังใจ และให้คำแนะนำต่างๆที่เป็นประโยชน์ยิ่งจนทำให้โครงการนี้สำเร็จลุล่วงไปได้ด้วยดี

เพื่อน ๆ ทุกท่านที่คอยให้คำแนะนำ ให้ความช่วยเหลือ และเป็นกำลังใจมาโดยตลอด

จึงใคร่ขอขอบคุณบุคคลดังกล่าวข้างต้นมา ณ โอกาสนี้

นันทวัฒน์ ไชยรัตน์

สารบัญ

สารบัญ (ต่อ)

หน้า

สารบัญตาราง

สารบัญรูป

รูปที่ หน้า

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ปัจจุบันเทคโนโลยีเกี่ยวกับระบบอินเทอร์เน็ตมีความก้าวหน้ามากขึ้นทำให้ทุกคนเข้าสู่โลกของอินเทอร์เน็ตได้มากขึ้น ด้วยราคาค่าบริการที่ถูกลง และความเร็วในการให้บริการที่เพิ่มขึ้น ซึ่งผู้ใช้งานส่วนใหญ่จะใช้บริการอินเทอร์เน็ตในการรับหรือหาข้อมูลและความบันเทิงจากเว็บไซต์ ด้วยเหตุผลเดียวกันทำให้มีผู้ให้บริการเว็บไซต์เกิดขึ้นมากมาย แน่นอนว่าจำนวนข้อมูลก็เพิ่มขึ้นทุกวันเช่นเดียวกัน ซึ่งจะไม่มีประโยชน์เลยหากผู้ใช้งานไม่สามารถเข้าถึงข้อมูลเหล่านั้นได้ วิธีการแก้ปัญหาที่ดีที่สุดในปัจจุบันคือพึ่งพาความสามารถของระบบเว็บเสิร์ชเอนจิน (Search Engine) ในการค้นหาเว็บไซต์ที่ตรงกับความต้องการของผู้ใช้งานโดยอาศัยสื่อกลางคือ คำที่ใช้ในการค้นหา (Key Word) ซึ่งแน่นอนว่าคำหนึ่งคำย่อมมีหลายความหมาย ทำให้ผู้ใช้งานได้รับผลลัพธ์จากการค้นหาได้หลากหลายเช่นเดียวกันความหมายของคำ ทำให้การเข้าถึงข้อมูลเกิดความล่าช้าเพราะต้องหาผลลัทธ์ที่ต้องการจากผลลัพท์จำนวนมาก ซึ่งไม่สามารถรับประกันได้ว่าเว็บไซต์ที่ได้สามารถใช้งานได้หรือไม่ และในบางกรณีหากต้องเนื้อหาเดียวกันจากเว็บไซต์อื่นๆ เพื่อการอ้างอิง หรือเพื่อให้แน่ใจในข้อมูลที่ได้รับ ก็อาจจะต้องใช้เวลาในหาเปรียบเทียบเนื้อหาจากเว็บไซต์ที่ได้ทั้งหมด

บทความนี้ต้องการศึกษาระบบการค้นหาแบบเจาะจง (Focus Web Search) เพื่อพัฒนาผลลัทธ์ที่ได้จากระบบเว็บเสิร์ชเอนจินที่ได้รับความนิยมทั่วไป โดยมุ่งเน้นไปที่การแก้ปัญหาที่ได้กล่าวมาแล้วในข้างต้น การใช้ระบบการค้นหาแบบเจาะจง จะทำการหาเว็บไซต์ที่เหมือนกับเว็บไซต์ที่ผู้ใช้งานได้เลือกไว้ ทำให้ผู้ใช้งานได้ผลลัพธ์เป็นเว็บไซต์ที่ตรงกับความต้องการของผู้ใช้งานได้มากขึ้นและประหยัดเวลา รวมทั้งระบบจะตรวจสอบก่อนว่าเว็บไซต์ที่ได้นั้นยังใช้งานได้อยู่หรือไม่

1.2 วัตถุประสงค์ของโครงการ

วัตถุประสงค์ของโครงการนี้เพื่อเป็นแนวทางในการออกแบบและพัฒนาระบบการค้นหาแบบเจาะจง (Specific Web Search) และศึกษาการทำงานของเสิร์ชเอนจิน (Search Engine) โดยใช้เทคโนโลยีการทำเว็บไมนิ่ง (Web Mining) เข้ามาประยุกต์ใช้เพื่อวัตถุประสงค์ต่อไปนี้

1.2.1 เพื่อศึกษาระบบการทำงานของเสิร์ชเอนจิน (Search Engine)

1.2.2 ศึกษาการทำงานของเว็บไมนิ่ง และ การดึงข้อมูลออกจากเอกสาร (Information Retrieval) เพื่อนำมาใช้ในการหาเว็บไซต์ที่มีความเหมือนกัน

1.2.3 เพื่อพัฒนาระบบการการค้นหาแบบเจาะจง (Specific Web Search)

1.2.4 เพื่อพัฒนาผลลัทธ์ที่ได้จากเสิร์ชเอนจิน (Search Engine) ให้ตรงกับความต้องการของผู้ใช้งานมากขึ้น , ลดเวลาที่ใช้ในการค้นหาเว็บไซต์ และได้ผลลัทธ์เพิ่มขึ้น โดยใช้ระบบการค้นหาแบบเจาะจง

1.2.5 เพื่อหาเว็บไซต์ที่มีความเหมือนกัน

1.3 ขอบเขตของโครงการ

วัตถุประสงค์ของโครงการนี้เพื่อศึกษาการทำงานของเสิร์ชเอนจิน และพัฒนาระบบการค้นหาแบบเจาะจง (Specific Web Search) ทำให้แบ่งขอบเขตของการศึกษาและพัฒนาได้เป็น ... ส่วนดังนี้

1.3.1 ศึกษาการทำงานของเสิร์ชเอนจินศึกษาว่ามีการทำงานอย่างไร โดยมีขั้นตอนดังต่อไปนี้

1.3.1.1 การเก็บข้อมูลจากเว็บไซต์

1.3.1.2 การดึงเนื้อหาที่ได้จากข้อมูลที่ได้

1.3.1.3 การทำสารบัญข้อมูล เพื่ออ้างอิงถึงข้อมูลที่เก็บมาได้ และเพื่อสามารถเรียกใช้งานได้

1.3.1.4 การเรียงลำดับเว็บไซต์ที่มีความสัมพันธ์กับคำที่ใช้ค้นหามากที่สุด

1.3.2 ศึกษาการทำงานของอินโฟสไปเดอร์ และนำมาประยุกต์ให้เป็นระบบการการค้นหาแบบเจาะจง (Specific Web Search) โดยขั้นตอนที่เพิ่มขึ้นจากการทำงานของเสิร์ชเอนจินแบบปรกติมีดังนี้

1.3.2.1 การหาลิงก์ที่เกี่ยวข้องกับเนื้อหาที่อยู่ในเว็บไซต์ที่ได้เก็บข้อมูลมา

1.3.2.2 การหาความเหมือนกันของเว็บไซต์

1.3.3 พัฒนาระบบการการค้นหาแบบเจาะจง (Specific Web Search) โดยมีขั้นตอนในการทำงานดังนี้

1.3.3.1 ผู้ใช้งานค้นหาเว็บไซต์โดยใช้คำที่ใช้ในการค้นหา (Key word)

1.3.3.2 ระบบจะนำคำค้นที่ได้ไปค้นหากับเว็บเสิร์ชเอนจินทั่วไปในที่นี้จะใช้บริการของ Bing API เพื่อให้ได้เว็บไซต์ตั้งต้นที่เกี่ยวข้องต่อไป

1.3.3.3 ผู้ใช้งานจะเลือกเว็บไซต์ที่ผู้ใช้งานคิดว่าตรงกับความต้องการ จากนั้นระบบจะทำการหาเว็บไซต์ที่มีความเหมือนกับเว็บไซต์ตั้งต้นที่ผู้ใช้ได้เลือกไว้ โดยค้นหาจากลิงก์ที่อยู่ในเว็บไซต์ตั้งต้นทำให้ได้เว็บไซต์ใหม่ที่อยู่นอกเหนือจากผลลัทธ์ของบริง (Bing Search Engine API)

1.4 ประโยชน์ที่คาดว่าจะได้รับ

1.4.1 ประโยชน์ต่อผู้ทำการพัฒนาระบบ

1.4.1.1 เป็นการเรียนรู้การทำงานของระบบเสิร์ชเอนจิน โดยคาดหวังที่จะนำความรู้ไปประยุกต์ใช้ในอนาคต

1.4.1.2 เป็นการเรียนรู้การทำงานของอินโฟสไปเดอร์ และวิธีการที่ใช้ในการดึงข้อมูลออกจากเอกสาร (Information Retrieval) จากเอกสารหรือข้อมูลจากเว็บไซต์ต่างๆ เพื่อนำมาใช้ประโยชน์ในการประมวลผลต่อไป

1.4.1.3 รู้จักวิเคราะห์และออกแบบระบบเว็บแอพลิเคชั่น และการพัฒนาระบบอย่างมีประสิทธิภาพ

1.4.1.4 รู้จักวิธีการวางแผนการพัฒนา และการแก้ปัญหาที่เกิดขึ้นในการพัฒนาระบบ

1.4.2 ประโยชน์ต่อผู้ใช้งานระบบ

1.4.2.1 เพิ่มประสิทธิภาพของผลลัพธ์ที่ได้จากการค้นหาของ Bing Search Engine โดยผลลัพธ์ที่ได้จากการค้นหาแบบเจาะจง จะเพิ่มขึ้นจากผลลัทธ์เดิมของ Bing Search Engine และตรงกับความต้องการของผู้ใช้งานมากขึ้น

1.4.2.2 ผู้ใช้งานสามารถค้นหาเว็บไซต์ที่เหมือนกับเว็บไซต์ที่เลือกไว้ได้โดยไม่ต้องเสียเวลาค้นหาเอง

1.4.2.3 ผู้ใช้สามารถรู้ได้ว่าเว็บไซต์ที่เป็นผลลัทธ์จาก Bing Search Engine ยังสามารถใช้งานได้อยู่หรือไม่ โดยไม่ต้องคลิกเข้าไปดูเอง

1.5 ขั้นตอนในการพัฒนาระบบงาน

ขั้นตอนในการพัฒนาระบบงานจะประกอบด้วยขั้นตอนดังต่อไปนี้

1.5.1 ศึกษาระบบการทำงานของเสิร์ชเอนจิน

1.5.2 ศึกษาการทำงานของอินโฟสไปเดอร์

1.5.3 ศึกษาวิธีการดึงข้อมูลออกจากเอกสาร (Information Retrieval) จากเอกสารและเว็บเพจ

1.5.4 นำข้อมูลที่ได้มาออกแบบระบบการค้นหาแบบเจาะจง (Specific Search)

1.5.5 พัฒนาระบบตามข้อมูลที่ได้ทำการวิเคราะห์

1.5.6 ทดสอบระบบเพื่อหาการตั้งค่าที่เหมาะสม เพื่อให้ได้ผลลัพธ์จากการค้นหาแบบเจาะจง (Specific Search) ที่ดีที่สุด

1.5.7 สรุปผลการทดสอบระบบเพื่อเปรียบเทียบผลลัพธ์กับการตั้งค่าที่ได้ และเปรียบเทียบกับผลลัพธ์ของบริง (Bing Search Engine)

บทที่ 2
ทฤษฎีและหลักการที่ใช้ในการพัฒนาระบบงาน

2.1 การทำงานของเสิร์ชเอนจิน

หลักการทำงานของเสิร์ชเอนจิน จะมีโปรแกรมที่เรียกว่า "คอลเลอร์" หรือ "โรบอท" ใช้ในการเข้าไปเก็บข้อมูลที่เว็บไซต์ต่างๆ จากนั้นก็จำนำข้อมูลเหล่านั้นมากำจัดคำซึ่งเป็น "นอยส์" ออกไป หรือเรียกว่าการเตรียมข้อมูล (Pre-processing text) จากนั้นก็จะผ่านกระบวนการในการทำสารบัญข้อมูล และเก็บลงสู่ฐานข้อมูล เมื่อผู้ใช้งานส่งคีเวอร์ เข้ามาก็จะนำไปเปรียบเทียบและแสดงผลต่อไป โดยส่วนใหญ่การเก็บข้อมูลสารบัญข้อมูล จะเก็บเป็น URLs และคำโดยการจะให้ได้ผลลัพธ์ที่ตรงกับที่ผู้ใช้ต้องการ สามารถแบ่งขั้นตอนการทำเสิร์สเอนจินได้เป็น 2 ขั้นตอนดังนี้ "คอลเลอร์" และ "แรงค์กิ้ง"

คอลเลอร์ จะทำหน้าที่ในการสร้างสารบัญข้อมูล เพื่อให้ตัว อัลกอลิทึมของเสิร์ชเอนจิน ดึงเอกสารที่ได้รับการทำสารบัญข้อมูล แล้วซึ่งมีความสัมพันธ์กันขึ้นมา เพื่อนำไปประมวลผลต่อไปในขบวนการของการทำแรงค์กิ้ง ซึ่งจะพูดถึงต่อไป

แรงค์กิ้ง จะทำให้หน้าที่ในการบอกว่าเอกสารที่ดึงขึ้นมานั้นมีความสัมพันธ์กับคำที่ค้นหรือไม่ และให้คะแนนเพื่อใช้จัดลำดับความสัมพันธ์ได้

จะสังเกตุได้ว่าการทำสารบัญข้อมูล นั้นเป็นการทำล่วงหน้าก่อนที่จะมีการค้นหาข้อมูล ซึ่งในความเป็นจริงแล้วนั้นโดยลักษณะของเว็บไซต์นั้นมีความเปลี่ยนแปลงสูงมาก ทำให้การทำ สารบัญข้อมูล ล่วงหน้าอาจมีความผิดพลาดได้ เช่นหากเว็บไซต์ที่นำมาทำสารบัญข้อมูล นั้นมีการเปลี่ยนแปลงเนื้อหา หรือ ลบเนื้อหาในส่วนนั้นไปแล้ว ดังนั้นจึงควรทำสารบัญข้อมูล ให้บ่อยๆเพื่อลดความผิดพลาด แต่การสารบัญข้อมูล ก็ไม่สามารถทำได้บ่อยๆ เนื่องจากข้อมูลเว็บไซต์นั้นมีมากทำให้การทำสารบัญข้อมูล นั้นต้องใช้เวลาและทรัพยากรสูง

ปัญหาที่ได้กล่าวมากสามารถแก้ไขได้โดย หากใช้ ออนไลน์เอเจน เพื่อเข้าไปดูเนื้อหาที่ได้จากการค้นหาของเสิร์ชเอนจินก่อนว่ามีการเปลี่ยนแปลงหรือไม่

นอกจากนี้จากการศึกษาพบว่า ในทุกๆเว็บไซต์ที่มีความสัมพันธ์กันจะมีลิงก์ที่เชื่อมโยงถึงกันโดยมีค่าเฉลี่ยอยู่ที่ 19 ลิงก์ (Filippo Menczer 2546:2) ซึ่งจะทำให้ออนไลน์เอเจนที่ส่งเข้าไปเก็บข้อมูล สามารถดึงเอาเว็บไซต์ที่มีความสัมพันธ์กันมาเพิ่มเป็นผลลัพธ์ในการค้นหาได้ในเวลาจำกัดโดยไม่จำเป็นต้องทำสารบัญข้อมูลใหม่

2.2 อินโฟสไปเดอร์

อินโฟสไปเดอร์ เป็นออนไลน์เอเจนชนิดหนึ่ง ซึ่งนำมาประยุกต์ใช้กับลักษณะของเว็บไซต์ที่มีความสัมพันธ์กันระหว่างเว็บเพจ มีข้อมูลที่ไม่ต้องการปะปนอยู่กับข้อมูลที่ต้องการ รวมทั้งข้อมูลที่ต้องการไม่ได้เก็บอยู่ในที่เดียว และมีการเปลี่ยนแปลงบ่อยบทความนี้จะประยุกต์ใช้อินโฟสไปเดอร์ในการท่องไปตามลิงก์ที่อยู่ในเว็บเพจ เพื่อหาเอกสารใหม่ที่เกี่ยวข้องกับเอกสารเดิม หรือเกี่ยวข้องกับคำที่ผู้ใช้ใช้ในการค้นหา ผลลัพธ์ที่ได้จากการใช้อินโฟสไปเดอร์ คือเอกสารใหม่เพิ่มเติม ที่มีความสดใหม่ไม่ใช่ลิงก์ที่เสียแล้ว และมีการเรียงลำดับความเหมือนกับเอกสารเดิม ที่ใช้เป็นต้นแบบในการค้น

การทำงานของอินโฟสไปเดอร์ นั้นจะเริ่มสร้างเอเจน ขึ้นมาเพื่อเรียนรู้ชุดเอกสารตั้งต้นโดยประเมินค่าของลิงก์ที่อยู่ในเอกสาร และเก็บข้อมูลที่อยู่ในเอกสาร(เว็บเพจ) ดังรูปที่ 1 โดยเอกสารตั้งต้นนั้นจะเป็นชุดของลิงก์ (URLs) ที่ชี้ไปยังหน้าที่มีความสัมพันธ์กับคิวรีเทอม (Query Term) ที่ผู้ใช้ใส่มาในระบบ ซึ่งมาจากการใช้คิวรีเทอม Query Term นั้นค้นไปที่เสิร์ชเอนจิน เพื่อให้ได้ชุดของลิงก์ (URLs) หรือ เป็นบุคมาร์ก (Bookmarks) ที่ผู้ใช้งานได้เก็บไว้เองก็ได้ โดยจำนวนนั้นขึ้นอยู่กับผู้ใช้งานเองว่าจะกำหนดเอกสารตั้งต้นแต่ละจุดจะมีเอเจน อยู่โดยแต่ละจุดของเอกสารจะมีเอ็นเอนร์จี (Energy) ซึ่งเป็นค่าตั้งต้นอยู่จำนวนหนึ่งเอ็นเอนร์จี (Energy) เป็นค่าที่ใช้ในการระบุว่า เอเจน ควรจะทำการเก็บเนื้อหานั้นต่อไป หรือควรจะหยุด ทุกครั้งที่เอเจนตัดสินใจที่จะตามลิงก์ไปที่เอกสารใหม่จะมีการคำนวนค่าเอ็นเอนร์จี (Energy) ใหม่โดยใช้การประเมินค่าของเอกสารนั้นกับคิวรีเทอมว่ามีความเหมือนกันมาน้อยเพียงใด หากเหมือนกันมากค่าเอ็นเอนร์จี (Energy) ก็จะเพิ่มขึ้น เพื่อให้สามารถค้นหาลิงก์เพิ่มเติมจากเอกสารใหม่ได้อีก ในส่วนนี้จะใช้ Cosine Similarity ในการประเมินค่า ดังสมการที่ 2.1 ถ้าไม่มีเอ็นเอนร์จี (Energy) มีเป็นตัววัดเอเจน จะเก็บไปเรื่อยๆ ไม่มีจุดสิ้นสุด

$$sim(q,p) = \frac{\sum_{k \in q \cap p} f_{kq} f_{kp}}{\sqrt{\left(\sum_{k \in p} f_{kp}^2\right)\left(\sum_{k \in q} f_{kp}^2\right)}} \qquad (2.1)$$

q คือ Query

p คือ เอกสารหน้าใหม่

f_{kp} คือความถี่ที่เจอ keyword ในเอกสาร p

หน้าที่หลักของ Agent คือประเมินว่าลิงก์ที่อยู่ในเอกสารนั้นมีคุณภาพมากเพียงใดโดยประเมินจากเนื้อหาที่อยู่ใกล้เคียงลิงก์นั้น ไม่ได้ประเมินจากเนื้อหาของเอกสารทั้งหมด

2.3 การดึงข้อมูลออกจากเอกสาร (Information Retrieval)

เป็นวิธีการที่ใช้ในการค้นหาเอกสาร หรือข้อมูลที่อยู่ในเอกสาร โดยในที่นี้เอกสารที่ต้องการค้นหาคือเว็บไซต์ ต่อไปนี้คือลำดับขั้นตอนในการทำการดึงข้อมูลออกจากเอกสาร(Information Retrieval) เพื่อใช้ในการค้นหาเว็บไซต์จากคำที่ใช้ในการค้นหา

2.3.1 การดึงข้อมูลออกจากเอกสาร (Documents Representation)

หลังจากที่ใช้ Agent เก็บข้อมูลที่เป็นจากเว็บเพจมาเรียบร้อยแล้ว หลังจากนั้นจะทำการลบคำสั่งที่เป็น HTML ออกไปจากข้อมูล ลบคำที่ปรากฏบ่อยที่ไม่ได้ช่วยบ่งบอกถึงความหมายของเอกสารนั้น หรือไม่มีความหมาย (StopWords) หากเป็นภาษาอังกฤษจะทำ Stemmer เพื่อทำให้คำนั้นอยู่ในรูปพื้นฐานเช่น is , are จะแปลงเป็น be หลังจากที่ผ่านขั้นตอนที่ได้กล่าวมาทั้งหมดเอกสารที่ได้จะอยู่ในรูปของคำ (Term) เป็นคำที่อยู่ในเอกสารเรียงต่อกันโดยตารางที่ 2.1 แสดงค่าเปรียบเทียบระหว่างค่าของคำที่ได้จากเอกสารให้ครั้งแรก (Word) และ คำที่ผ่านการแปลงแล้ว (Term) หลังจากที่ได้ Term เรียบร้อยแล้วจะทำสร้างตาราง Document Representation ซึ่งสามารถสร้างได้ 2 วิธีดังนี้

2.3.1.1 Boolean Term-Document Matrix

ดังตารางที่ 2.2 จะเป็นการนำ Term ของเอกสารทั้งหมดดังตัวอย่างเป็นเอกสาร 20 ฉบับที่มี Term ทั้งหมดอยู่ 5 Term โดยตารางจะบอกว่าเอกสารฉบับใดมี Term ใดอยู่บ้าง

จากตารางที่ 2.2 เอกสาร d1 มี Term คือ program อยู่เพียงคำเดียวเท่านั้น การทำ Boolean Term เป็นวิธีที่สามารถทำได้ง่ายเพราะบอกแต่ว่าเอกสารนั้นมี Term อยู่ค่าเป็น 1 หรือไม่มี ค่าเป็น 0 แต่การทำไปใช้เพื่อบ่งบอกถึงเอกลักษณ์ของเอกสารนั้นไม่สามารถทำได้ดีมากนัก เนื่องจากการเก็บข้อมูลมีเพียงใช่หรือไม่ใช่ เมื่อนำไปใช้งานจริง ในการค้นหาเอกสารที่มี Term ที่ตรงกับ Query Term จะได้ผลลัพธ์เอกสารจำนวนมาก ไม่เหมาะกับการนำมาใช้กับการค้นหาข้อมูลจากเว็บไซต์ซึ่งมีข้อมูลจำนวนมาก อีกทั้ง Term ที่อยู่ในเว็บเพจอาจมีเหมือนกันแต่ไม่ใช่เอกสารที่กล่าวถึงเนื้อหาเรื่องเดียวกันก็ได้ดังนั้นจึงมีอีกวิธีเพื่อจัดการปัญหาที่กล่าวมา

2.3.1.2 Term-Document Matrix with Term Position

ดังตารางที่ 3 วิธีการเบื้องต้นเหมือนกับ Boolean แต่การเก็บค่าในตารางนั้นจะเก็บตำแหน่งของ Term ที่ปรากฏในเอกสารด้วย ดังในตารางเอกสาร d3 จะมี Term ของ laboratory อยู่ในลำดับที่ 65 และ 69 และ computer อยู่ลำดับที่ 68

ข้อมูลที่อยู่ในตารางนั้นมีจำนวนมากจนไม่นิยมเก็บไว้ในฐานข้อมูลเชิงสัมพันธ์ ดังนั้นวิธีการที่นิยมใช้คือ เก็บไว้ในรูปแบบไฟล์ โดยอาจเก็บโดยใช้ B-trees หรือ Hash table

ตารางที่ 2.1 แสดงการเปรียบเทียบระหว่าง Word และ Term

Document ID	Document Name	Words	Terms
d_1	Anthropology	114	86
d_2	Art	153	105
d_3	Biology	123	91
d_4	Chemistry	87	58
d_5	Communication	124	88
d_6	Computer Science	101	77
d_7	Criminal Justice	85	60
d_8	Economics	107	76
d_9	English	116	80
d_{10}	Geography	95	68
d_{11}	History	108	78
d_{12}	Mathematics	89	66
d_{13}	Modern Languages	110	75
d_{14}	Music	137	91
d_{15}	Philosophy	85	54
d_{16}	Physics	130	100
d_{17}	Political Science	120	86
d_{18}	Psychology	96	60
d_{19}	Sociology	99	66
d_{20}	Theatre	116	80
Total number of words/terms		2195	1545
Number of different words/terms		744	671

ตารางที่ 2.2 แสดงการทำ Document Representation แบบ Boolean Term

Document ID	*lab*	*laboratory*	*programming*	*computer*	*program*
d_1	0	0	0	0	1
d_2	0	0	0	0	1
d_3	0	1	0	1	0
d_4	0	0	0	1	1
d_5	0	0	0	0	0
d_6	0	0	1	1	1
d_7	0	0	0	0	1
d_8	0	0	0	0	1
d_9	0	0	0	0	0
d_{10}	0	0	0	0	0
d_{11}	0	0	0	0	0
d_{12}	0	0	0	1	0
d_{13}	0	0	0	0	0
d_{14}	1	0	0	1	1
d_{15}	0	0	0	0	1
d_{16}	0	0	0	0	1
d_{17}	0	0	0	0	1
d_{18}	0	0	0	0	0
d_{19}	0	0	0	0	1
d_{20}	0	0	0	0	0

ตารางที่ 2.3 แสดงการทำ Document Representation แบบ Matrix โดยจะเก็บตำแหน่งของ Term ที่อยู่ในเอกสารด้วย

Document ID	*lab*	*laboratory*	*programming*	*computer*	*program*
d_1	0	0	0	0	[71]
d_2	0	0	0	0	[7]
d_3	0	[65,69]	0	[68]	0
d_4	0	0	0	[26]	[30,43]
d_5	0	0	0	0	0
d_6	0	0	[40,42]	[1,3,7,13,26,34]	[11,18,61]
d_7	0	0	0	0	[9,42]
d_8	0	0	0	0	[57]
d_9	0	0	0	0	0
d_{10}	0	0	0	0	0
d_{11}	0	0	0	0	0
d_{12}	0	0	0	[17]	0
d_{13}	0	0	0	0	0
d_{14}	[42]	0	0	[41]	[71]
d_{15}	0	0	0	0	[37,38]
d_{16}	0	0	0	0	[81]
d_{17}	0	0	0	0	[68]
d_{18}	0	0	0	0	0
d_{19}	0	0	0	0	[51]
d_{20}	0	0	0	0	0

2.3.2 การหาค่าความถี่ของคำที่ปรากฏ (Term Frequency (TF))

เอกสารที่ได้มาเป็นจำนวนมากนั้นประกอบด้วย Term มากมายจะรู้ได้อย่างไรว่าเอกสารได้สัมพันธ์กับ Query Term หรือจะเรียงลำดับความสัมพันธ์ของเอกสารกับคำที่ใช้ในการสืบค้นได้อย่างไร ในส่วนนี้จะใช้ความถี่ของ Term ที่อยู่ในเอกสารเป็นตัววัดโดยจะแปลงเอกสารให้เป็น Vertor ที่อยู่ใน Euclidean Space โดยการแปลงจะใช้ค่าความถี่ของ Term ที่ได้จากตารางที่ 2.3 โดยวิธีการหาความถี่ของคำที่ปรากฏมีทั้งหมด 3 วิธีดังนี้

กำหนดให้ n เป็นจำนวนเอกสาร m เป็น term ที่อยู่ในเอกสารนั้น nij เป็นจำนวนครั้งที่ term ti ปรากฏในเอกสาร

2.3.2.1 ใช้จำนวน Term ที่ต้องการหา หารด้วย Term ทั้งหมดที่อยู่ในเอกสารดังสมการ ที่ 2.2

$$TF(t_i, d_j) = \begin{cases} 0 & if\ n_{ij} = 0 \\ \frac{n_{ij}}{\sum_{k=1}^{m} n_{kj}} & if\ n_{ij} > 0 \end{cases} \quad (2.2)$$

จากข้อมูลที่กล่าวมาในข้างต้นค่า IDF ของ Term ที่ i โดยจะนำจำนวนครั้งที่ปรากฏของ Term ทั้งหมด (1,545 ครั้ง) หารด้วย จำนวนครั้งที่ Term ที่ i ปรากฏ หากคำนวนตามตัวอย่างที่ได้จะได้ค่า IDF ของทั้ง 6 Term ดังตารางที่ 2.4

ตารางที่ 2.4 แสดงการเปรียบเทียบค่า IDF ของทั้ง 6 Term

lab	*laboratory*	*programming*	*computer*	*program*
3.04452	3.04452	3.04452	1.43508	0.559616

จากตารางที่ 2.4 เห็นได้ว่าค่า IDF ขึ้นอยู่กับจำนวนเอกสารที่ปรากฏ ใน 3 Term แรก คือ lab , laboratory , programming มีค่า IDF มากเมื่อเทียบกับอีก 2 Term ที่เหลือก หากสังเกตดูจากตารางที่ 2.3 จะเห็นว่าแต่ละ Term นั้นมีการปรากฏที่เอกสารฉบับเดียวเนื่องจากความคิดที่ว่า Term ที่มีปรากฏอยู่น้อยในนั้นน่าจะมีความสำคัญกับเอกสารมากว่า Term อื่นๆที่มีปรากฏอยู่ในหลายๆ เอกสาร อาจเป็นคำเฉพาะที่สามารถสื่อถึงเอกสารนั้น หากลองคำนวนค่า TFIDF จากในเอกสารที่ 6 จะได้ค่าดังนี้

ค่า TF จากเอกสาร D6

$\overrightarrow{D_6}$ = (0 0 0.026 0.078 0.039)

ค่า TFIDF จากเอกสาร D6

$\overrightarrow{D_6}$ = (0 0 0.079 0.112 0.022)

จากค่าที่ได้แสดงให้เห็นว่า Computer ยังเป็นคำที่สำคัญกับเอกสารมากที่สุดแต่คำที่เพิ่มลำดับความสำคัญสื่อถึงเอกสารนี้ขึ้นมาคือ Programming

2.4 การเรียงลำดับความสัมพันธ์ของเอกสารกับคำที่ใช้ในการสืบค้น (Document Ranking)

การเรียงลำดับเอกสารที่มีความสัมพันธ์กับคำที่ใช้ในการสืบค้น Query Term ในขั้นแรกจะทำการแปลง Query Term ให้อยู่ในรูปแบบเวอร์เตอร์ ดังตัวอย่าง

QUERY TERM = {COMPUTER , PROGRAM}

ค่า TF ของ Query Term

$\vec{q}$ = (0 0 0 0.5 0.5)

จากนั้นจะทำการปรับให้เป็นค่า TFIDF โดยคูณค่า IDF ที่ได้จากการคำนวนในหัวข้อที่ผ่านมาได้ดังตัวอย่าง

ค่า TFIDF ของ Query Term
$\vec{q}$ = (0 0 0 0.7175 0.2798)

นำค่า TFIDF ของ Query Term ไปเปรียบเทียบกับค่า TFIDF ของเอกสารทั้งหมดที่ได้สร้างไว้แล้ว โดยมีวิธีการเปรียบเทียบ 2 วิธีดังนี้

2.4.1 Vector Difference เป็นเปรียบเทียบโดยวัดระยะห่างระหว่าง Vector ของเอกสารที่ต้องการวัด กับ Query Vector ว่ามีความห่างกันเพียงใด โดยค่าความต่างที่น้อยจะแสดงให้เห็นว่าเอกสารนั้นมีความเกี่ยวข้องกับ Query Term มากโดยให้ความสำคัญกับเอกสารที่มี Query Term ครบทั้ง 2 คำก่อนเนื่องจากเอกสารที่มี Query Term มีคำเดียวจะมีค่าน้อยกว่า โดยแสดงดังสมการที่ 2.6

$$\|\vec{q} - \vec{d_j}\| = \sqrt{\sum_{i=1}^{m} (q^i - d_j^i)^2} \tag{2.6}$$

2.4.2 Cosine Similarity ใช้วิธีนำ Document vector ไป dot กับ Query Vector โดยผลลัพธ์ที่ได้หากค่ามากแสดงว่าเอกสารนั้นมีความใกล้เคียงกับ Query Term มากโดยจะให้ความสำคัญกับเอกสารที่มี Query Term ครบทั้ง 2 คำก่อนเช่นเดียวกัน ดังสมการที่ 2.7

$$\vec{q} \cdot \vec{d_j} = \sum_{i=1}^{m} q^i d_j^i \tag{2.7}$$

2.5 การเปรียบเทียบความเหมือนกันของเอกสาร

การเปรียบเทียบความเหมือนของเอกสาร ใช้เพื่อเปรียบเทียบเอกสารตั้งต้น กับเอกสารใหม่ที่ได้มาจากลิงก์ในเอกสารนั้น เพื่อเปรียบเทียบว่าควรดึงลิงก์ที่อยู่ในเอกสารนั้นขึ้นมาอีกหรือไม่ หรือควรจะนำเอกสารใหม่ที่ได้ไปประมวลผลต่อไปหรือไม่ วิธีในการคำนวนจะใช้วิธี Cosine Similarity เช่นเดียวกัน โดยการหา Term ที่มีใช้ในการคำนวนนั้นสามารถแบ่งได้ทั้งหมด 4 วิธีคือ

1.เปรียบเทียบโดยใช้ Term ทั้งหมดที่ปรากฏในเอกสารทุกเอกสารที่มีเปรียบเทียบกับเอกสารตั้งต้น เป็นวิธีที่ง่ายที่สุดแต่เสียเวลาในการประมวลผลมากเพราะจำนวน Term ที่มีอยู่มีจำนวนมาก

2.เปรียบเทียบโดยใช้ Term ที่ปรากฏมากที่สุดนั่นก็คือใช้ชุดของ Term ที่มีค่า TF มากที่สุดเพื่อใช้เป็นตัวแทนของเอกสารนั้นในการนำมาเปรียบเทียบกับ Term ที่มีค่า TF มากที่สุดของเอกสารตั้งต้นเช่นเดียวกัน ปัญหาของวิธีนี้อยู่ที่ค่า TF ที่มากอาจเป็นไปได้ว่าคำนั้นเป็นคำที่มีความหมายกว้างๆ ทำให้อาจมีปรากฏอยู่ในหลายๆเอกสารเช่นเดียวกัน หากเลือกชุดของ Term น้อยเกินไปก็อาจได้ผลลัพธ์ของเอกสารที่มีความเหมือนกันเป็นจำนวนมาก

3.เปรียบเทียบโดยใช้ชุดของ Term ที่มีค่า IDF มากที่สุดมาเปรียบเทียบกัน วิธีนี้จะทำให้ได้เอกลักษณ์ของเอกสารมาเปรียบเทียบกันได้ดี แต่จะมีปัญหาอยู่ที่ค่า IDF จะมากก็ต่อเมื่อมีเอกสารที่มี Term นี้อยู่น้อยจึงจำเป็นต้องใช้หลายๆ Term มาประมวลผล

4.เปรียบเทียบโดยใช้ใช้ชุดของ Term ที่มีค่า TFIDF มากที่สุดมาเปรียบเทียบกัน ในบทความนี้จะใช้วิธีนี้ในการหาค่าความเหมือนกันของเอกสาร

ปัญหาของการหาเอกสารที่มีความเหมือนกัน คือจำนวน Term ที่นำมาเปรียบเทียบควรจะมีจำนวนเท่าไรจึงจะทำให้ได้ผลลัพธ์ที่เหมาะสม และใช้เวลาในการประมวลผลน้อย นอกจากนี้จำนวน Term ที่นำมาใช้ในการเปรียบเทียบมีผลต่อผลลัพธ์ที่ได้อีกด้วย การใช้จำนวน Term ที่ต่างกันในการเปรียบเทียบอาจได้ลำดับของเอกสารที่มีความเหมือนกันแตกต่างกันออกไป

บทที่ 3
การวิเคราะห์ระบบปัจจุบัน

ระบบการค้นหาของเสิร์ชเอนจินในปัจจุบันมีขั้นตอนในการทำงานตั้งแต่เริ่มเก็บข้อมูลตามเว็บไซต์ต่างๆ จากนั้นนำข้อมูลมารวมกันและทำการประมวลผลเพิ่มสร้างสารบัญคำที่ใช้ในการค้นหาขึ้นมา หากจะแบ่งก็จะมีขั้นตอนหลักๆ ดังต่อไปนี้

3.1 ขั้นตอนการทำงานของเสิร์ชเอนจินในปัจจุบัน

การทำงานของเสิร์ชเอนจินนั้นสามารถแบ่งได้ออกเป็น 2 ขั้นตอนหลักๆคือ การเก็บข้อมูลเว็บไซต์ และ การเรียงลำดับความสัมพันธ์กับคำค้น โดยมีหลักการในการทำงานดังนี้

1.การเก็บข้อมูลเว็บไซต์ เป็นขั้นตอนในการเก็บข้อมูลจากเว็บไซต์ต่างๆ เพื่อนำมาเก็บไว้ในฐานข้อมูล โดยเก็บข้อมูลที่ได้พร้อมกับชื่อของเว็บไซต์นั้น พร้อมกับลิงก์ต่างๆที่เชื่อโยงกัน จัดทำเป็นสารบัญข้อมูลเพื่อให้เข้าถึงและเรียกใช้ได้ง่าย โดยผ่านขั้นตอนตามที่ได้กล่าวในบทที่ 2

2.การเรียงลำดับความสัมพันธ์กับคำค้น เมื่อผู้ใช้ได้เข้าใช้งานเสิร์ชเอนจิน ก็ต้องเริ่มจากการกรอกคำค้นเข้าไปในระบบ หลังจากนั้นระบบจะดำเนินขั้นตอนในการเรียงลำดับความสัมพันธ์ของคำที่ใช้ในการค้นหากับสารบัญข้อมูลเว็บไซต์ได้ที่มาจากขั้นตอนแรก

จากรูปที่ 3.1 แสดงให้เห็นขั้นตอนในการทำงานของระบบเสิร์ชเอนจินโดยส่วน Indexer robot จะทำหน้าที่เก็บข้อมูลเว็บไซต์มาทำเป็นสารบัญเว็บไซต์เพื่อให้ระบบ Search Engine ทำหน้าที่เรียงลำดับความสัมพันธ์กับคำค้นของผู้ใช้ต่อไป

3.2 ปัญหาที่พบในการทำงานของเสิร์ชเอนจิน

ปัญหาระบบเสิร์ชเอนจินนั้นมาจากจำนวนข้อมูลมหาศาลที่อยู่ในระบบ ซึ่งข้อมูลนี้ก็มาจากจำนวนเว็บไซต์ที่เพิ่มขึ้นทุกวันรวมถึงจำนวนผู้ใช้งานที่มากขึ้นทำให้มีข้อมูลในระบบเพิ่มขึ้น โดยหากจะแบ่งเป็นหัวข้อหลักสามารถแบ่งได้ดังนี้

1. ปัญหาที่เกิดจากจำนวนข้อมูล ทำให้ขนาดของสารบัญเว็บไซต์มีขนาดใหญ่มาก ทำให้ต้องใช้ทรัพยากรณ์ในการประมวลผลข้อมูลส่วนนี้มากและใช้เวลานาน เป็นภาระให้กับระบบ ด้วยเหตุนี้เองทำให้การทำสารบัญไม่สามารถทำได้บ่อย

2. ด้วยลักษณะของเว็บไซต์มีลักษณะคือการเปลี่ยนแปลงบ่อย ไม่ว่าจะเป็นการเพิ่มข้อมูลในเว็บไซต์ การลบข้อมูล หรือการเปลี่ยนแปลงข้อมูล รวมทั้งจำนวนที่เพิ่มขึ้นอย่างมหาศาลของเว็บไซต์ในปัจจุบัน ทำให้การค้นหาข้อมูลที่เพิ่มขึ้นเหล่านี้ทำได้โดยลำบาก ต้องพึ่งพาระบบการเพิ่ม

บทที่ 4
การประยุกต์ใช้อินโฟสไปเดอร์เข้ากับระบบการค้นหาแบบเดิม

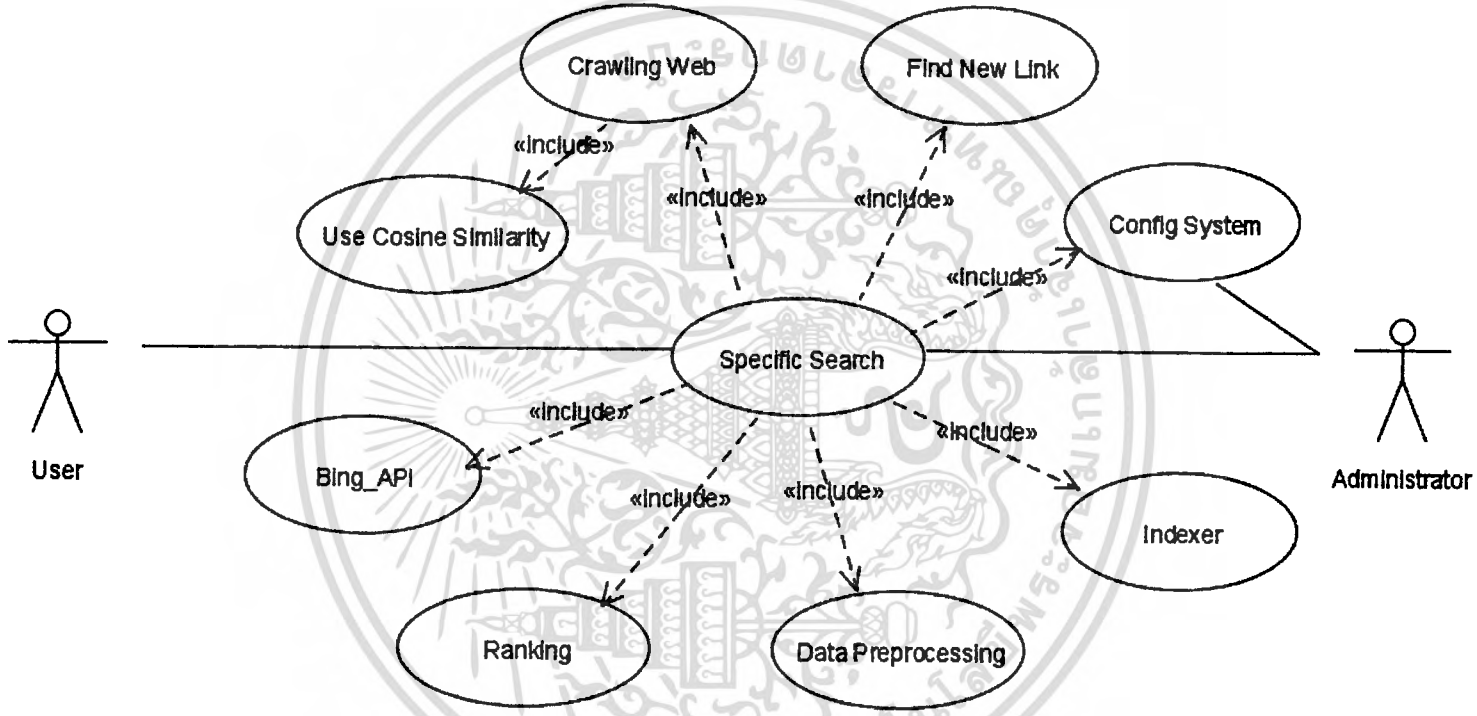
จากปัญหาของเสิรช์เอนจินในปัจจุบันคือ จำนวนข้อมูลที่เพิ่มขึ้นจนทำให้ระบบทำสารบัญเว็บไซต์ไม่สามารถทำงานได้ทันกับจำนวนของข้อมูลที่เพิ่มขึ้น ดังนั้นจึงต้องนำระบบของอินโฟสไปเดอร์เข้ามาปรับปรุงการทำงาน โดยเพิ่มเข้าไปในระบบเดิม โดยหลังจากที่ระบบเดิมได้ส่งผลการค้นหาเว็บไซต์ให้กับผู้ใช้เรียบร้อยแล้ว หลังจากนั้นผู้ใช้งานจะเลือกเว็บไซต์ที่ผู้ใช้คิดว่าตรงกับความต้องการ เพื่อให้ระบบอินโฟสไปเดอร์ทำงานหาผลลัพธ์เพิ่มเติมจากผลลัพธ์เดิม

4.1 ระบบการทำงาน

ระบบการทำงานของระบบการค้นหาเว็บไซต์แบบเจาะจงนั้นจะแบ่งได้เป็นส่วนของผู้ใช้งาน ผู้ดูแลระบบ และส่วนการทำงานของระบบ

โดยในส่วนของผู้ใช้งานนั้นจะใช้งานในการหาเว็บไซต์ที่ผู้ใช้ต้องการ โดยใช้คำค้น จากนั้นระบบจะส่งคำค้นที่ผู้ใช้ได้ใส่เข้ามาในระบบส่งไปยังระบบ Bing API เพื่อให้ได้ผลลัพธ์เป็นเว็บไซต์ที่เกี่ยวข้องกับคำที่ผู้ใช้ใช้ในการค้นหาจาก Bing API จากนั้นผู้ใช้ก็จะทำการเลือกเว็บไซต์ที่ผู้ใช้คิดว่าเป็นเว็บไซต์ที่ผู้ใช้งานสนใจกี่เว็บไซต์ก็ได้ จากนั้นระบบก็จะทำการหาเว็บไซต์ที่มีความเกี่ยวข้องกับเว็บไซต์ที่ผู้ใช้เลือก โดยเว็บไซต์ที่ได้นั้นเป็นเว็บไซต์ที่เป็นลิงก์ที่มาจากเว็บไซต์ที่ผู้ใช้ได้เลือกไว้

ส่วนของผู้ดูแลระบบจะทำการตั้งค่าเพื่อปรับให้ระบบสามารถทำงานได้อย่างมีประสิทธิภาพมากขึ้น แล้วแต่ตามชนิดของเว็บไซต์ที่ผู้ใช้ค้นหา รายละเอียดตามไดอะแกรมต่อไปนี้

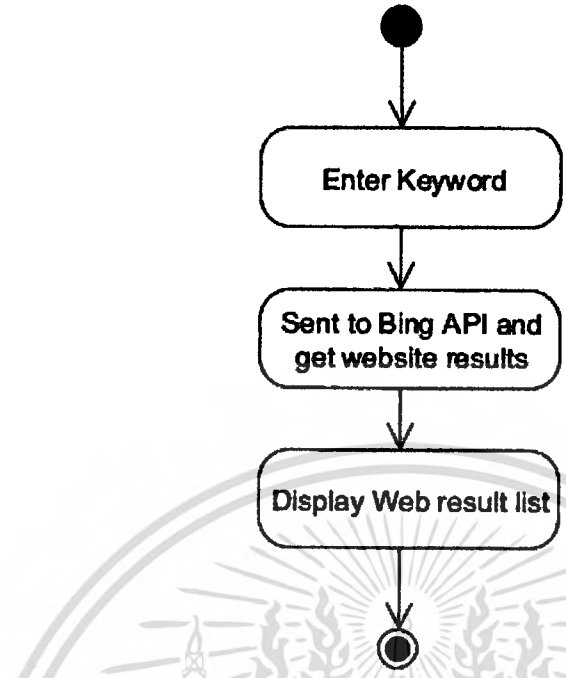


รูปที่ 4.1 ยูสเคสไดอะแกรมของระบบการค้นหาเว็บไซต์แบบเจาะจง

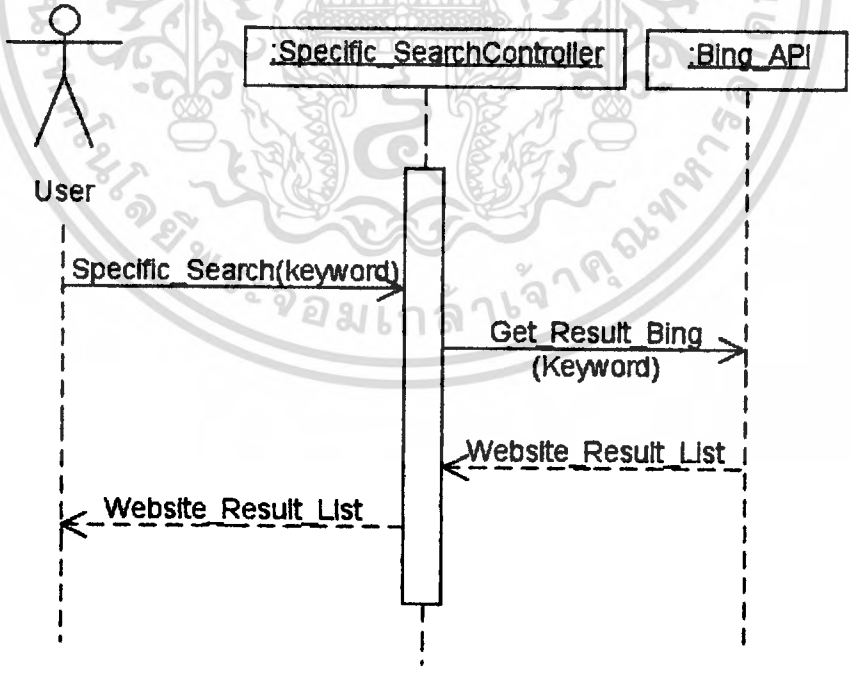
4.1.1 คำอธิบายยูสเคสไดอะแกรม

ตารางที่ 4.1 คำอธิบายยูสเคสไดอะแกรมของ Bing API

ยูสเคส	Bing API
วัตถุประสงค์	ค้นหาเว็บไซต์ที่ตรงกับความต้องการของผู้ใช้ โดยผ่าน Bing API และนำเว็บไซต์นั้นเป็นเว็บไซต์ที่ผู้ใช้เลือกเพื่อให้นำไปหาลิงก์เพิ่มเติมจากเว็บไซต์ที่ผู้ใช้เลือกต่อไป
เงื่อนไขเมื่อเริ่มต้น	ผู้ใช้งานใส่คำค้นที่ต้องการ
เมื่อทำงานสำเร็จ	ผู้ใช้งานมีผลลัพธ์เป็นเว็บไซต์ที่ใกล้เคียงกับคำที่ผู้ใช้ค้นหา โดยนำผลลัพธ์ที่ได้จาก Bing API มาแสดงผล
เมื่อทำงานไม่สำเร็จ	ผู้ใช้งานไม่สามารถใช้งานระบบได้ ระบบจะนำผู้ใช้งานไปที่หน้าแรกของเว็บไซต์
แอคเตอร์ที่เกี่ยวข้อง	User , Administrator
สิ่งที่กระตุ้นการทำงาน	ผู้ใช้ใส่คำค้นที่ต้องการไปในช่องค้นหา หลังจากนั้นผู้ใช้งานคลิกปุ่ม Search
อินพุต	คำค้น
เอาต์พุต	เว็บไซต์ที่ใกล้เคียงกับคำที่ใช้ค้นหา โดยเป็นเว็บไซต์ที่เป็นผลลัพธ์มาจาก Bing API
รายละเอียด	1. ผู้ใช้งานค้นหาเว็บไซต์ด้วยคำค้น 2. ระบบแสดงเว็บไซต์ที่เป็นผลลัพธ์ของคำค้น



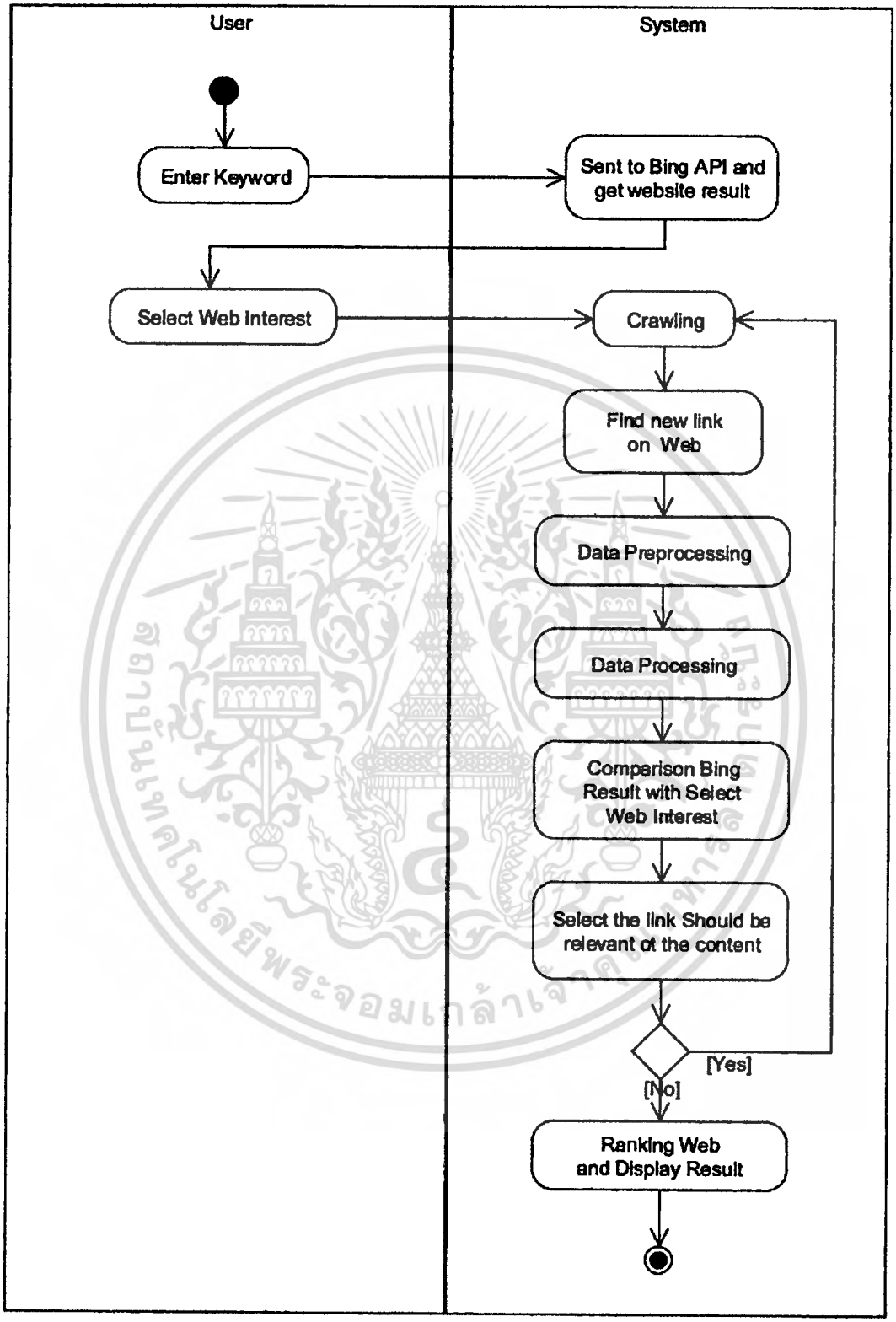
รูปที่ 4.2 แอ็กทิวิตีไดอะแกรมของระบบ Normal Search



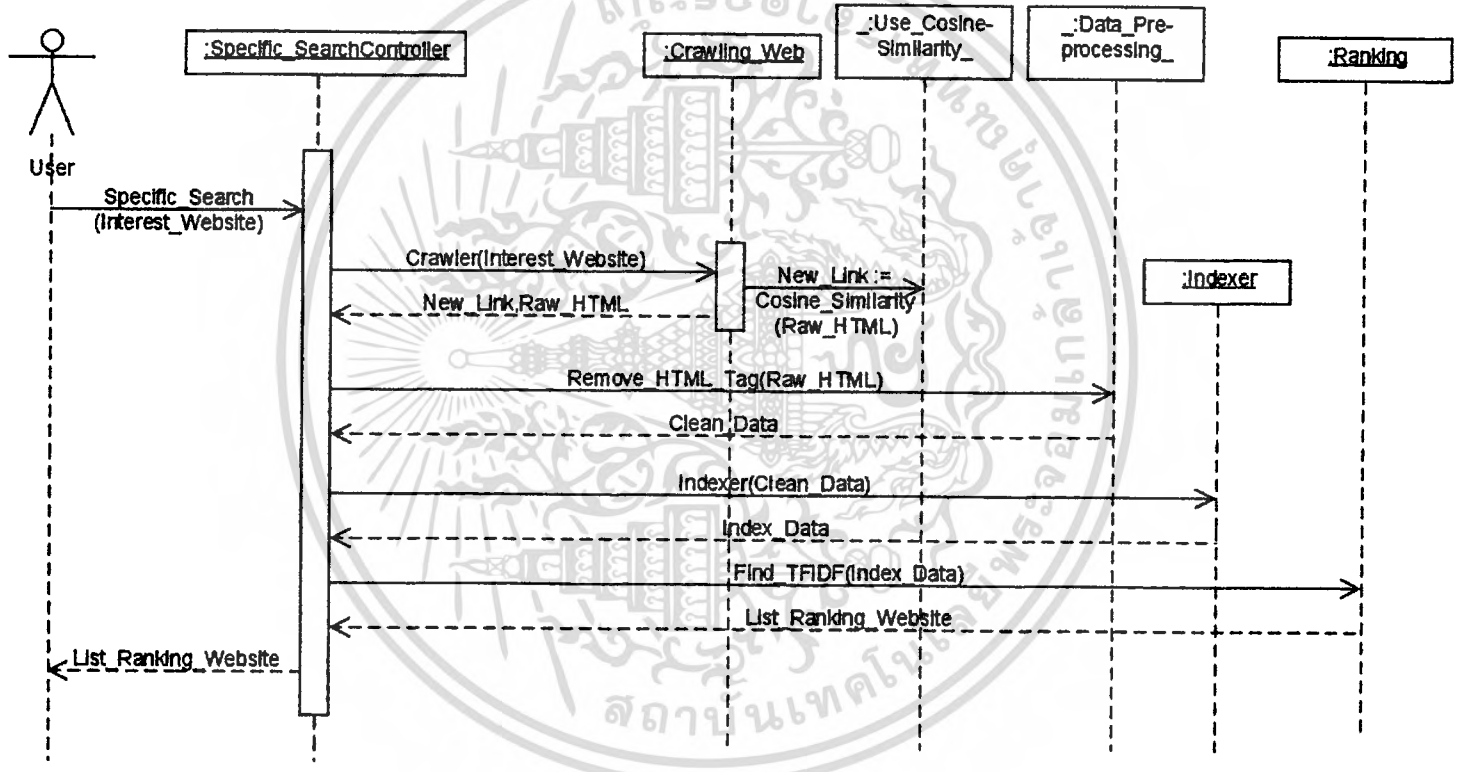
รูปที่ 4.3 ซีเควนซ์ไดอะแกรมของระบบ Bing API

ตารางที่ 4.2 คำอธิบายยูสเคสไดอะแกรมของ Specific Search

ยูสเคส	Specific Search
วัตถุประสงค์	เพื่อให้บริการค้นหาเว็บไซต์แบบเจาะจง ผู้ใช้จะได้เว็บไซต์ที่มีความเกี่ยวข้องกับเว็บไซต์ที่ผู้ใช้ได้เลือกไว้ โดยผลลัพธ์ที่ได้นั้นจะเป็นเว็บไซต์ที่อยู่ในเว็บไซต์ที่ผู้ใช้ได้เลือกไว้
เงื่อนไขเมื่อเริ่มต้น	ผู้ใช้ผ่านการค้นหาด้วยคำค้นมาจากขั้นตอน Bing API มาแล้ว และเลือกเว็บไซต์จำนวนหนึ่งจากผลลัพธ์ที่ Bing API แสดงขึ้นมาโดยเลือกที่ Check Box ด้านหน้าของเว็บไซต์ที่ผู้ใช้ต้องการ
เมื่อทำงานสำเร็จ	ผู้ใช้งานมีผลลัพธ์เป็นเว็บไซต์ที่เหมือนกับเว็บไซต์ที่ผู้ใช้งานเลือกไว้ โดยเป็นผลลัพธ์เพิ่มเติมจากผลการค้นหาแบบธรรมดา
เมื่อทำงานไม่สำเร็จ	ผู้ใช้งานไม่สามารถใช้งานระบบได้ ระบบจะนำผู้ใช้งานไปที่หน้าแรกของเว็บไซต์
แอคเตอร์ที่เกี่ยวข้อง	User , Administrator
สิ่งที่กระตุ้นการทำงาน	ผู้ใช้งานคลิกปุ่ม Specific Search
อินพุต	เว็บไซต์ที่ผู้ใช้งานสนใจ
เอาต์พุต	เว็บไซต์ที่เหมือนกับเว็บไซต์ที่ผู้ใช้งานสนใจ
รายละเอียด	1. ผู้ใช้งานค้นหาเว็บไซต์ด้วยคำค้น 2. ระบบแสดงเว็บไซต์ที่เป็นผลลัพธ์ของคำค้น 3. ผู้ใช้งานเลือกเว็บไซต์ที่สนใจจากผลลัพธ์ทั้งหมด 4. ระบบทำการแสดงเว็บไซต์ที่เหมือนกับเว็บไซต์ที่ผู้ใช้สนใจ โดยผลลัพธ์มาจากลิงก์ที่อยู่ในเว็บไซต์ที่ผู้ใช้ได้เลือกไว้



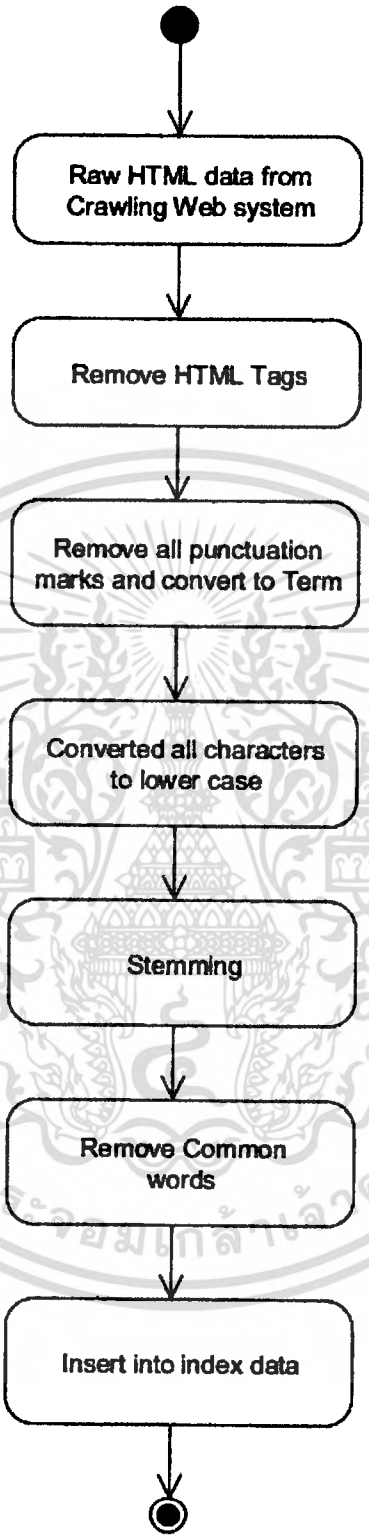
รูปที่ 4.4 แอ็กทิวิตีไดอะแกรมของระบบ Specific Search



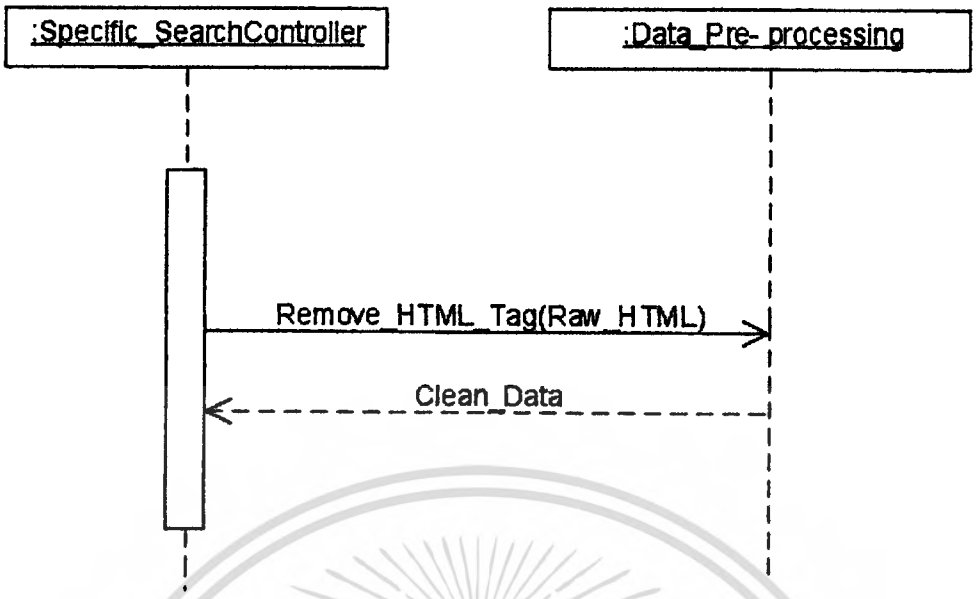
รูปที่ 4.5 ซีเควนซ์ไดอะแกรมของระบบ Specific Search

ตารางที่ 4.3 คำอธิบายยูสเคสไดอะแกรมของ Data Preprocessing

ยูสเคส	Data Preprocessing
วัตถุประสงค์	เพื่อให้เตรียมข้อมูลให้พร้อมกับการประมวลผล ในรูปแบบการค้นคืนข้อมูล (Information Retrieval)
เงื่อนไขเมื่อเริ่มต้น	ระบบทำการเก็บข้อมูลเว็บไซต์มาเป็นข้อมูลดิบ โดยที่ข้อมูลนี้ยังอยู่ในรูปของ HTML
เมื่อทำงานสำเร็จ	ข้อมูลที่อยู่ในรูปของสารบัญข้อมูล โดยจะคำที่ปรากฏอยู่ในเว็บไซต์ตามจำนวนความถี่ของคำ และทำการปรับค่าให้เป็นมาตราฐานเดียวกัน
เมื่อทำงานไม่สำเร็จ	ยกเลิกการเตรียมข้อมูล และไม่เก็บเว็บไซต์นั้นไว้พิจารณา
แอคเตอร์ที่เกี่ยวข้อง	System
สิ่งที่กระตุ้นการทำงาน	ระบบเก็บข้อมูลเว็บไซต์ส่งข้อมูลมา
อินพุต	ข้อมูลดิบที่อยู่ในรูปของ HTML
เอาต์พุต	ข้อมูลของเว็บไซต์นั้นๆที่อยู่ในรูปสารบัญความถี่คำ
รายละเอียด	1. ระบบเก็บข้อมูลเว็บไซต์มาในรูป HTML 2. ระบบทำการแปลงข้อมูลเหล่านั้นให้อยู่ในรูปสารบัญความถี่คำ



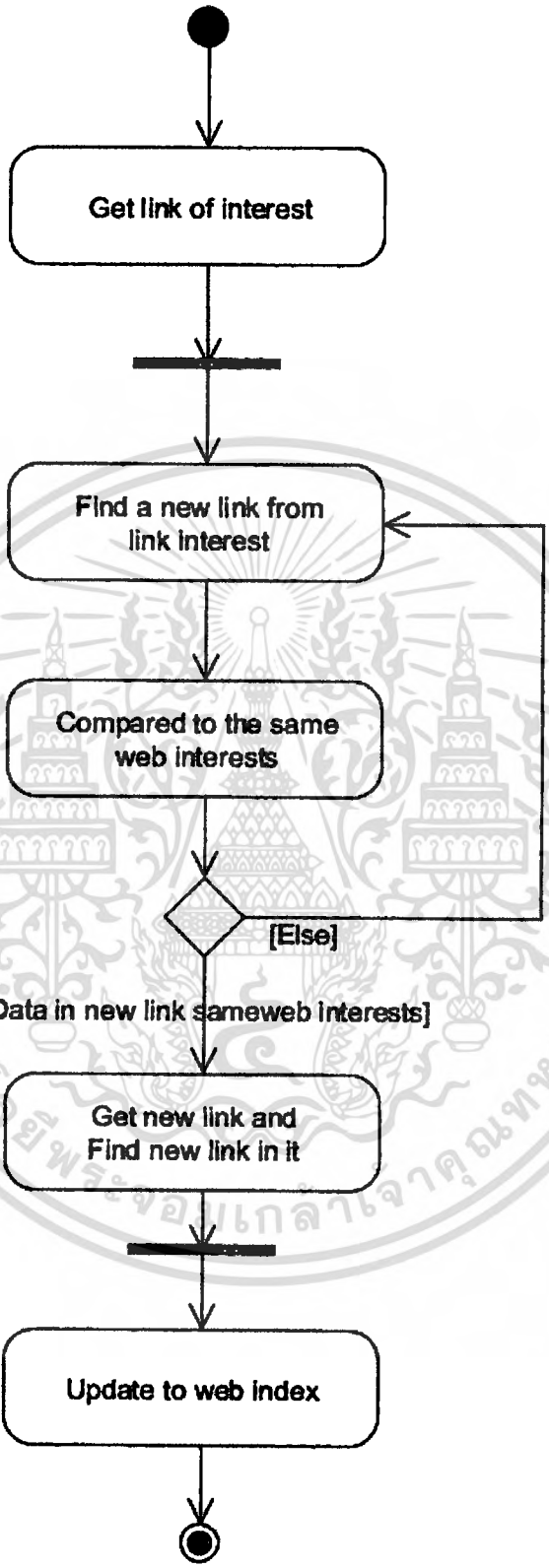
รูปที่ 4.6 แอ็กทิวิตีไดอะแกรมของระบบ Data Preprocessing



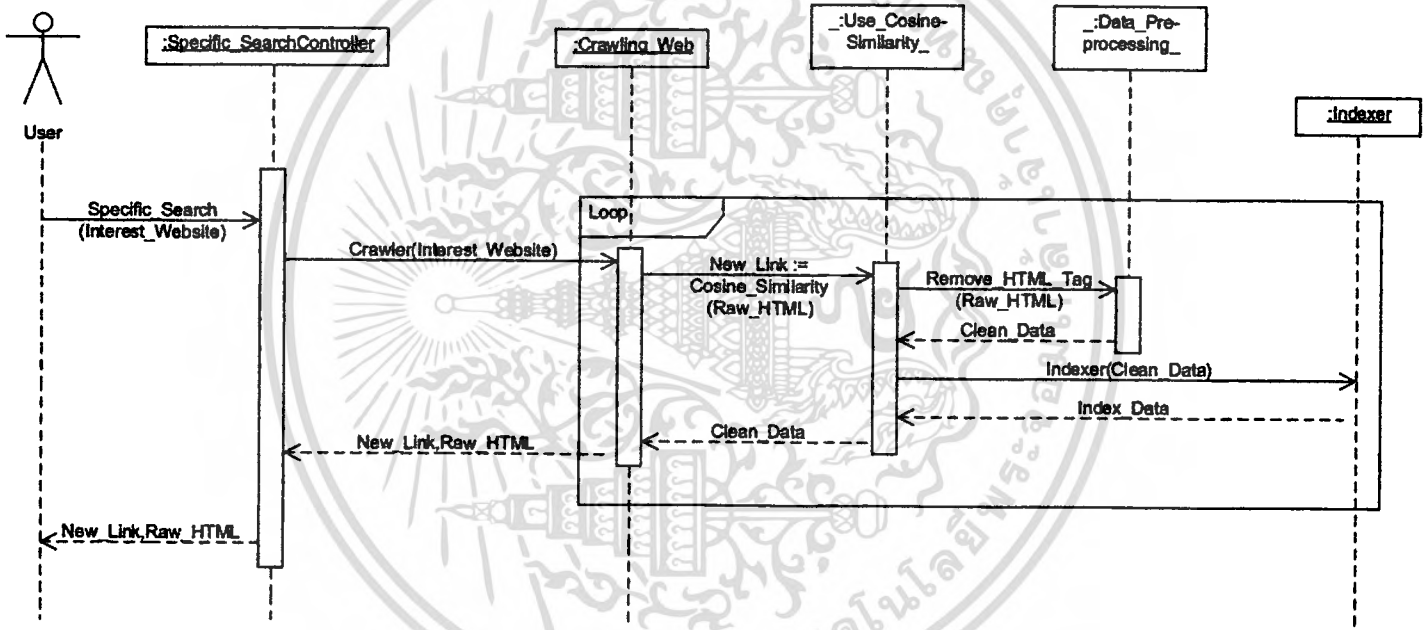
รูปที่ 4.7 ซีเควนซ์ไดอะแกรมของระบบ Bing API

ตารางที่ 4.4 คำอธิบายยูสเคสไดอะแกรมของ Crawling Web

ยูสเคส	Crawling Web
วัตถุประสงค์	เพื่อเก็บข้อมูลเว็บไซต์ในรูปแบบข้อมูล HTML ที่ระบบต้องการ โดยทำงานเป็นแบบ Multithread และสามารถเก็บข้อมูลเว็บไซต์ที่มีการ redirect ได้
เงื่อนไขเมื่อเริ่มต้น	ระบบส่งลิงก์ที่ต้องการให้เก็บข้อมูล
เมื่อทำงานสำเร็จ	ข้อมูลที่อยู่ในรูปของสารบัญข้อมูล โดยจะเก็บเป็นชื่อเว็บไซต์ และข้อมูลภายในเว็บไซต์ในรูปแบบ HTML
เมื่อทำงานไม่สำเร็จ	ยกเลิกการเก็บข้อมูล และไม่เก็บเว็บไซต์นั้นไว้พิจารณา
แอคเตอร์ที่เกี่ยวข้อง	System
สิ่งที่กระตุ้นการทำงาน	ระบบส่งลิงก์ที่ต้องการให้เก็บข้อมูล
อินพุต	ชื่อเว็บไซต์
เอาต์พุต	ชื่อเว็บไซต์พร้อมข้อมูลภายในเว็บในรูปแบบ HTML
รายละเอียด	1. ระบบส่งลิงก์ที่ต้องการให้เก็บข้อมูลเข้ามา 2. Crawling จะทำการเก็บข้อมูลในเว็บไซต์ที่ระบบต้องการ



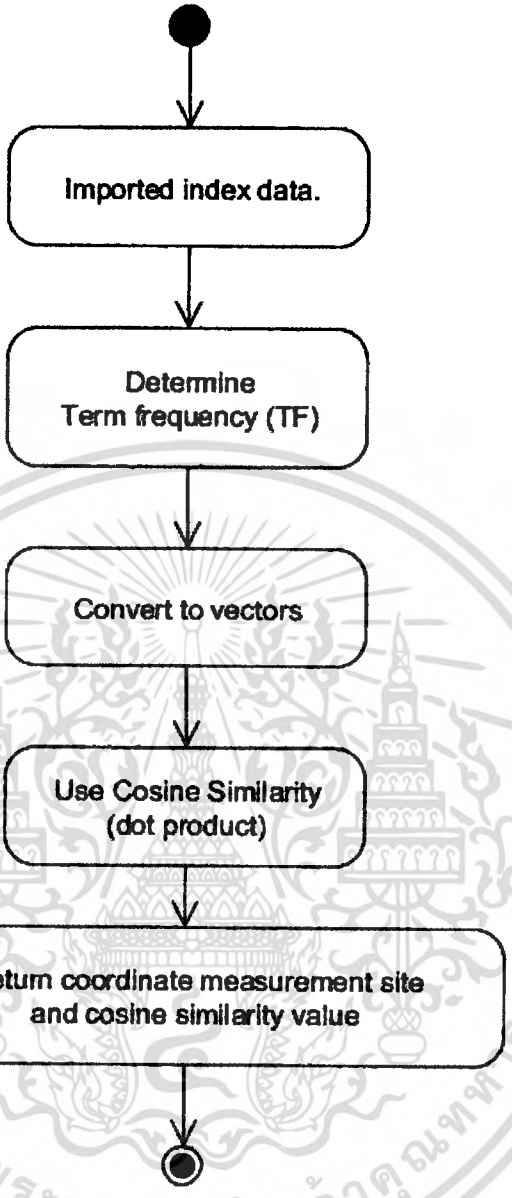
รูปที่ 4.8 แอ็กทิวิตีไดอะแกรมของระบบ Crawling Web



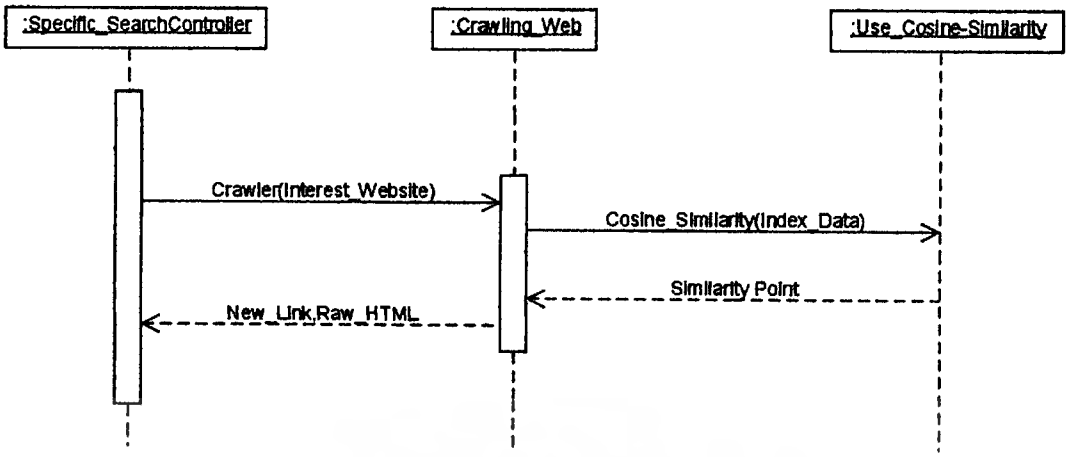
รูปที่ 4.9 ซีเควนซ์ไดอะแกรมของระบบ Crawling Web

ตารางที่ 4.5 คำอธิบายยูสเคสไดอะแกรมของ Use Cosine Similarity

ยูสเคส	Use Cosine Similarity
วัตถุประสงค์	เพื่อเปรียบเทียบความเหมือนกันของเว็บไซต์
เงื่อนไขเมื่อเริ่มต้น	ระบบส่งลิงก์ที่ต้องการให้เปรียบเทียบ
เมื่อทำงานสำเร็จ	ส่งข้อมูลออกมาในรูปของลิงก์ที่ทำการเปรียบเทียบ และค่าความเหมือนที่ได้
เมื่อทำงานไม่สำเร็จ	ให้ค่าความเหมือนที่ได้เป็น 0
แอคเตอร์ที่เกี่ยวข้อง	System
สิ่งที่กระตุ้นการทำงาน	ระบบส่งลิงก์ที่ต้องการให้เปรียบเทียบ
อินพุต	ชุดของเว็บไซต์ที่ต้องการให้เปรียบเทียบ
เอาต์พุต	ชื่อของเว็บไซต์ที่เปรียบเทียบและคะแนนที่ได้
รายละเอียด	1. ระบบส่งลิงก์ที่ต้องการให้เปรียบเทียบความเหมือน 2. ส่งข้อมูลกลับคืนระบบเป็นลิงก์ที่เปรียบเทียบ และค่าที่ได้



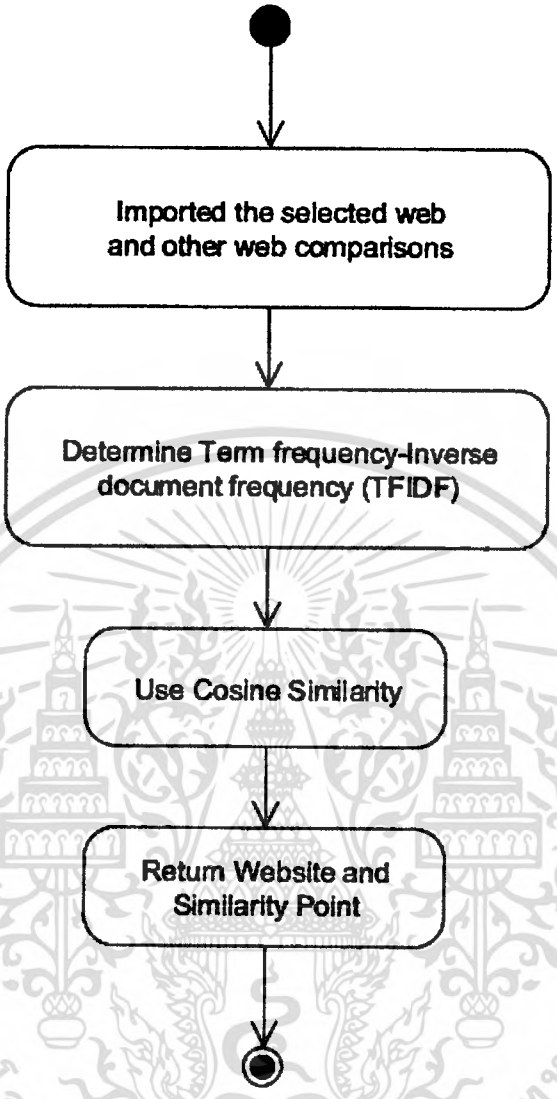
รูปที่ 4.10 แอ็กทิวิตีไดอะแกรมของระบบ Use Cosine Similarity



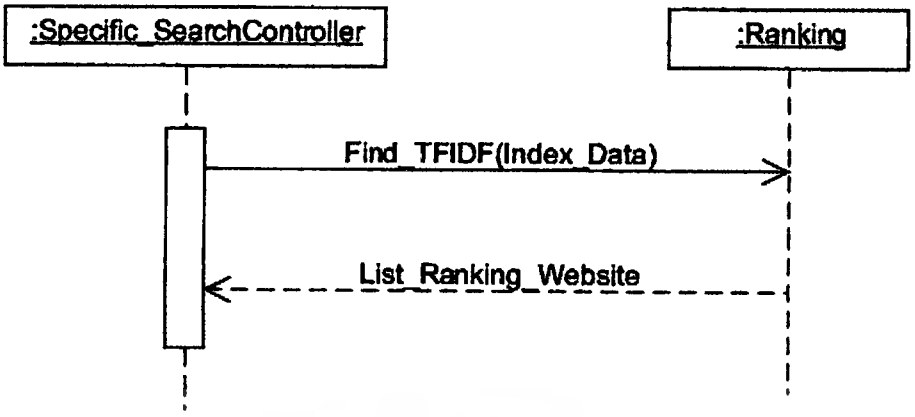
รูปที่ 4.11 ซีเควนซ์ไดอะแกรมของระบบ Use Cosine Similarity

ตารางที่ 4.6 คำอธิบายยูสเคสไดอะแกรมของ Ranking

ยูสเคส	Ranking
วัตถุประสงค์	เพื่อเรียงลำดับเว็บไซต์ที่มีความเหมือนกับเว็บไซต์ที่ผู้ใช้เลือกไว้มากที่สุด ไปยังน้อยที่สุด
เงื่อนไขเมื่อเริ่มต้น	ระบบส่งลิงก์ที่ต้องการให้เปรียบเทียบและเว็บไซต์ที่เลือก
เมื่อทำงานสำเร็จ	ส่งข้อมูลลำดับเว็บไซต์ที่มีความเหมือนกับเว็บไซต์ที่เลือก โดยบอกเป็นแต้ม
เมื่อทำงานไม่สำเร็จ	แจ้งว่าระบบไม่สามารถทำงานได้
แอคเตอร์ที่เกี่ยวข้อง	System
สิ่งที่กระตุ้นการทำงาน	ระบบส่งเว็บไซต์ที่เลือกไว้ และชุดเว็บไซต์ที่หาเพิ่มเติมได้
อินพุต	เว็บไซต์ที่เลือกไว้ และชุดของเว็บไซต์ที่ต้องการให้เปรียบเทียบ
เอาต์พุต	ชื่อของเว็บไซต์ที่เปรียบเทียบและคะแนนที่ได้
รายละเอียด	1. ระบบส่งลิงก์ที่ต้องการให้เปรียบเทียบความเหมือน 2. ส่งข้อมูลกลับคืนระบบเป็นลิงก์ที่เปรียบเทียบ และค่าที่ได้



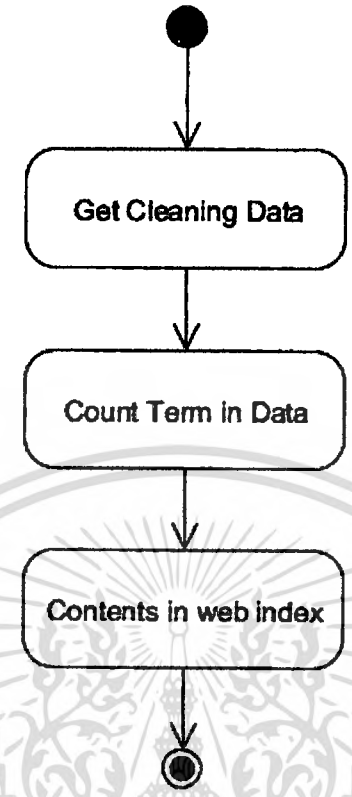
รูปที่ 4.12 แอ็กทิวิตีไดอะแกรมของระบบ Ranking



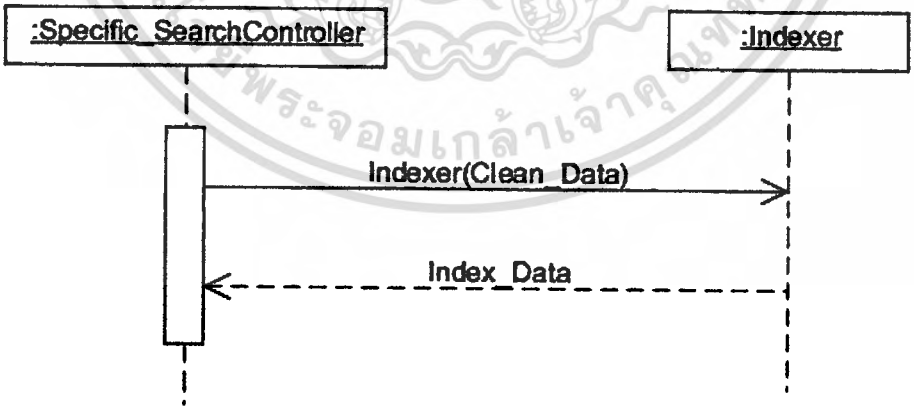
รูปที่ 4.13 ซีเควนซ์ไดอะแกรมของระบบ Ranking

ตารางที่ 4.7 คำอธิบายยูสเคสไดอะแกรมของ Indexer

ยูสเคส	Indexer
วัตถุประสงค์	เพื่อเก็บข้อมูลเว็บไซต์ให้อยู่ในรูปแบบที่พร้อมเรียกใช้เพื่อประมวลผลต่อไป
เงื่อนไขเมื่อเริ่มต้น	ระบบส่งข้อมูลเว็บไซต์ที่ผ่านการเตรียมข้อมูลแล้ว
เมื่อทำงานสำเร็จ	เก็บข้อมูลเว็บไซต์ในรูปของสารบัญเว็บไซต์
เมื่อทำงานไม่สำเร็จ	ยกเลิกการเก็บเว็บไซต์นั้นๆ
แอคเตอร์ที่เกี่ยวข้อง	System
สิ่งที่กระตุ้นการทำงาน	ระบบส่งข้อมูลเว็บไซต์ที่ผ่านการเตรียมข้อมูลแล้ว
อินพุต	ข้อมูลเว็บไซต์ที่ผ่านการเตรียมข้อมูลแล้ว
เอาต์พุต	ข้อมูลถูกแปลงให้อยู่ในรูปสารบัญเว็บไซต์
รายละเอียด	1. ระบบส่งข้อมูลที่ต้องการทำสารบัญข้อมูล 2. ข้อมูลที่ถูกส่งมาถูกแปลงให้อยู่ในรูปสารบัญเว็บไซต์



รูปที่ 4.14 แอ็กทิวิตีไดอะแกรมของระบบ Indexer

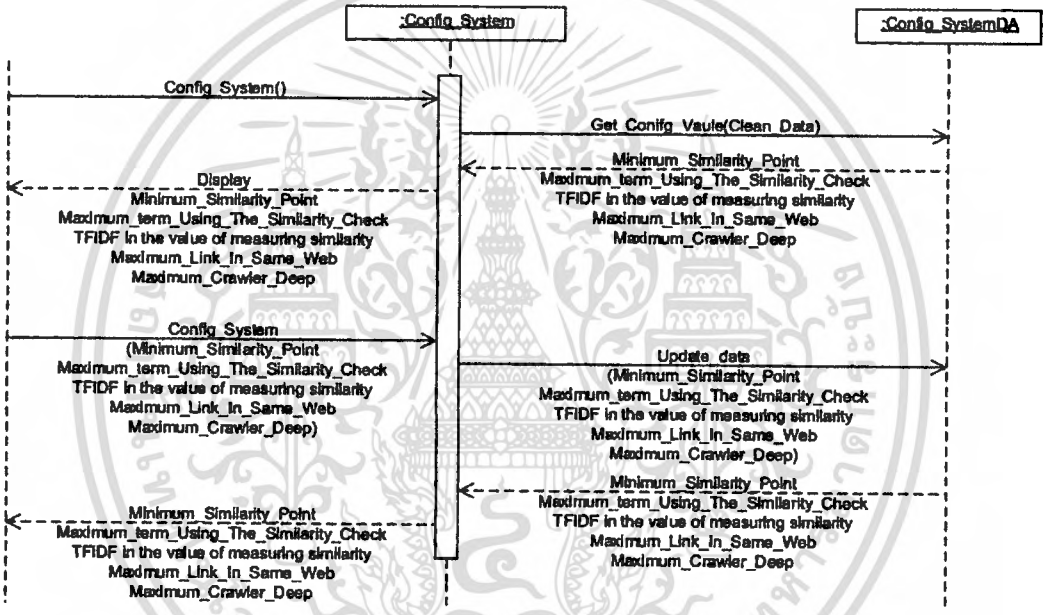


รูปที่ 4.15 ซีเควนซ์ไดอะแกรมของระบบ Indexer

ตารางที่ 4.8 คำอธิบายยูสเคสไดอะแกรมของ Config System

ยูสเคส	Config System
วัตถุประสงค์	เพื่อให้ผู้ดูแลระบบสามารถเข้าไปแก้ไขการทำงานของระบบ
เงื่อนไขเมื่อเริ่มต้น	ผู้ดูแลระบบแก้ไขข้อมูลระบบ
เมื่อทำงานสำเร็จ	ระบบเก็บค่าที่ผู้ดูแลระบบกำหนดไว้
เมื่อทำงานไม่สำเร็จ	เลือกใช้ค่าเดิมไม่มีการเปลี่ยนแปลงค่า
แอคเตอร์ที่เกี่ยวข้อง	Administrator
สิ่งที่กระตุ้นการทำงาน	ผู้ดูแลระบบคลิกที่ปุ่ม submit
อินพุต	1.ค่าความเหมือนต่ำสุด (Minimum similarity point) กำหนดเพื่อให้โปรแกรม ตัดสินใจในการหาเว็บไซต์เพิ่มเติมจากเว็บที่มีความเหมือนตามค่าที่กำหนด หรือใช้ในการจัดอันดับเว็บไซต์ผลลัพธ์ โดยหากมีค่าต่ำกว่าที่กำหนดจะไม่นำเว็บไซต์นั้นมาแสดงผล มีค่าได้ตั้งแต่ 0 - 100 2.ค่าคำมากสุดที่นำไปใช้ในการคำนวนหาค่าความเหมือนกันของเอกสาร (Maximim term using the similarity check) ค่านี้จะใช้กำหนดคำที่ใช้ในการคำนวนหาค่าความเหมือนโดยวิธีการ Cosine Similarity มีค่าตั้งแต่ 0 – 1000 3.กำหนดว่าต้องการให้โปรแกรมทำการแปลงค่าคำเป็น TF หรือ TFIDF เพื่อใช้ในการคำนวน (TFIDF in the value of measuring similarity) โดยกำหนดได้ทั้งหมด 2 ค่าคือ มีค่าเท่ากับ 1 ใช้ TFIDF ในการคำนวน มีค่าเท่ากับ 0 ใช้ค่า TF ในการคำนวน 4.ค่าลิงก์มากที่สุดที่โปรแกรมจะหาจากหนึ่งเว็บไซต์ (Maximum link in same web) ในหนึ่งเว็บไซต์มีลิงก์อยู่มากมายโปรแกรมจะไม่ทำการเปิดลิงก์ทั้งหมด แต่จะหาตามค่าที่ได้ทำการกำหนดไว้ มีค่าตั้งแต่ 1 – 50 5.ค่าความลึกมากสุดที่ให้โปรแกรมเข้าค้นหาเว็บไซต์ใหม่ (Maximum Crawler Deep) เมื่อโปรแกรมหาลิงก์ที่ได้จากเว็บไซต์ที่ผู้ใช้สนใจได้แล้ว ค่านี้จะกำหนดว่าจะให้หาลิงก์ใหม่จากเว็บใหม่ที่ได้หรือไม่ ถ้ามีค่าเป็น 1 คือไม่หา แต่ถ้ามีค่ามากกว่า 1 แสดงว่าให้หาเพิ่มตามจำนวนที่ได้

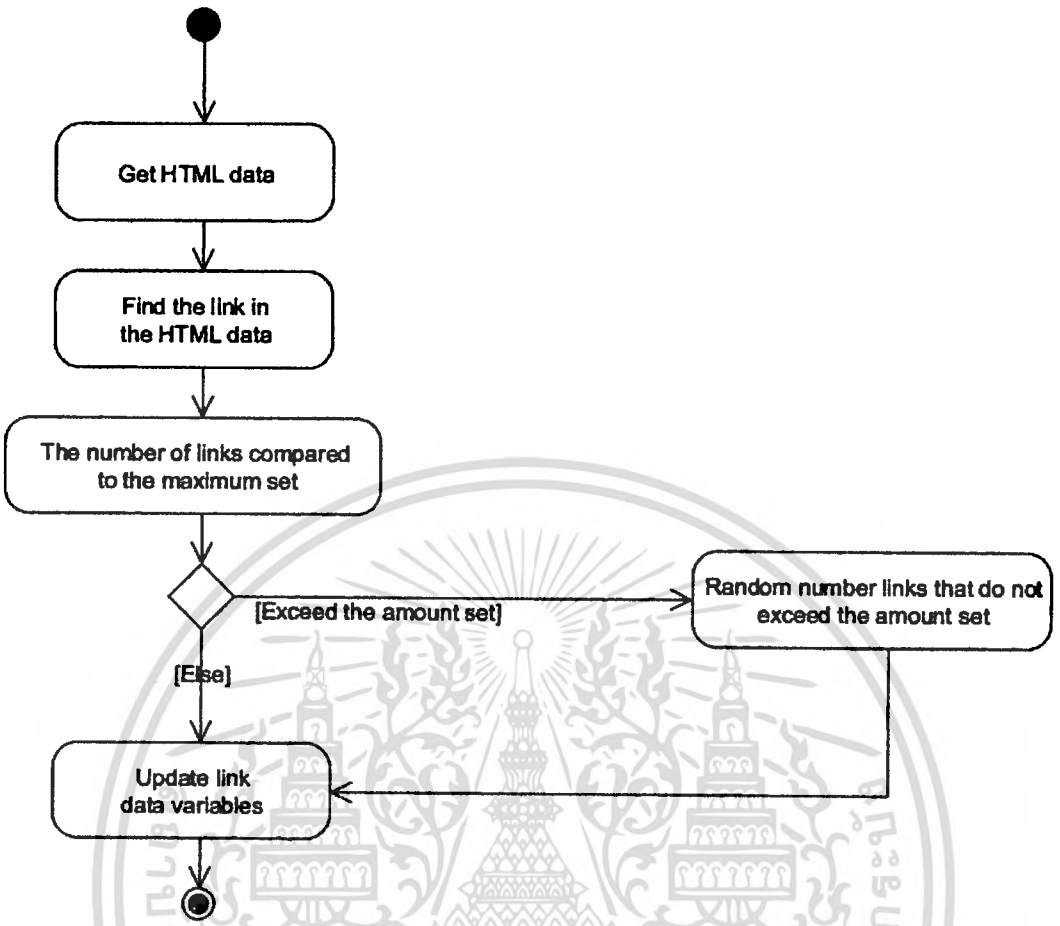
	มีค่าตั้งแต่ 1 – 10 ถ้ากำหนดมากกว่านี้อาจทำให้โปรแกรมค้างได้
เอาต์พุต	แสดงข้อมูลที่ผู้ดูแลระบบเปลี่ยนแปลง
รายละเอียด	1. ผู้ดูแลระบบแก้ไขข้อมูล 2. ระบบแสดงข้อมูลที่ทำการแก้ไข



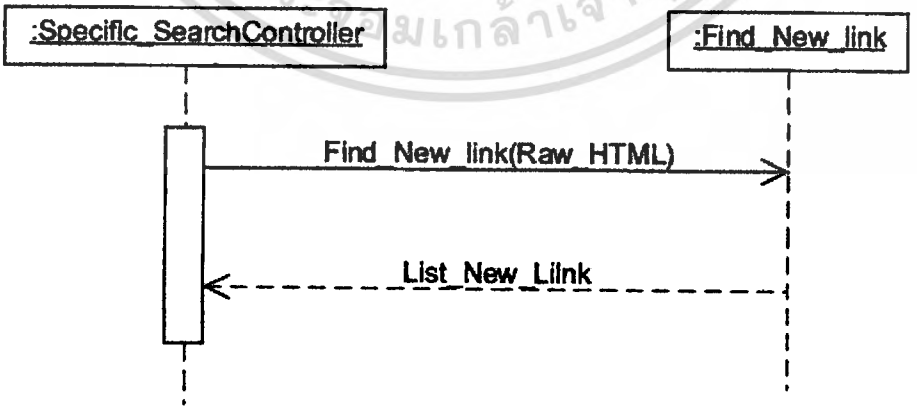
รูปที่ 4.16 ซีเควนซ์ไดอะแกรมของระบบ Config System

ตารางที่ 4.9 คำอธิบายยูสเคสไดอะแกรมของ Find New Link

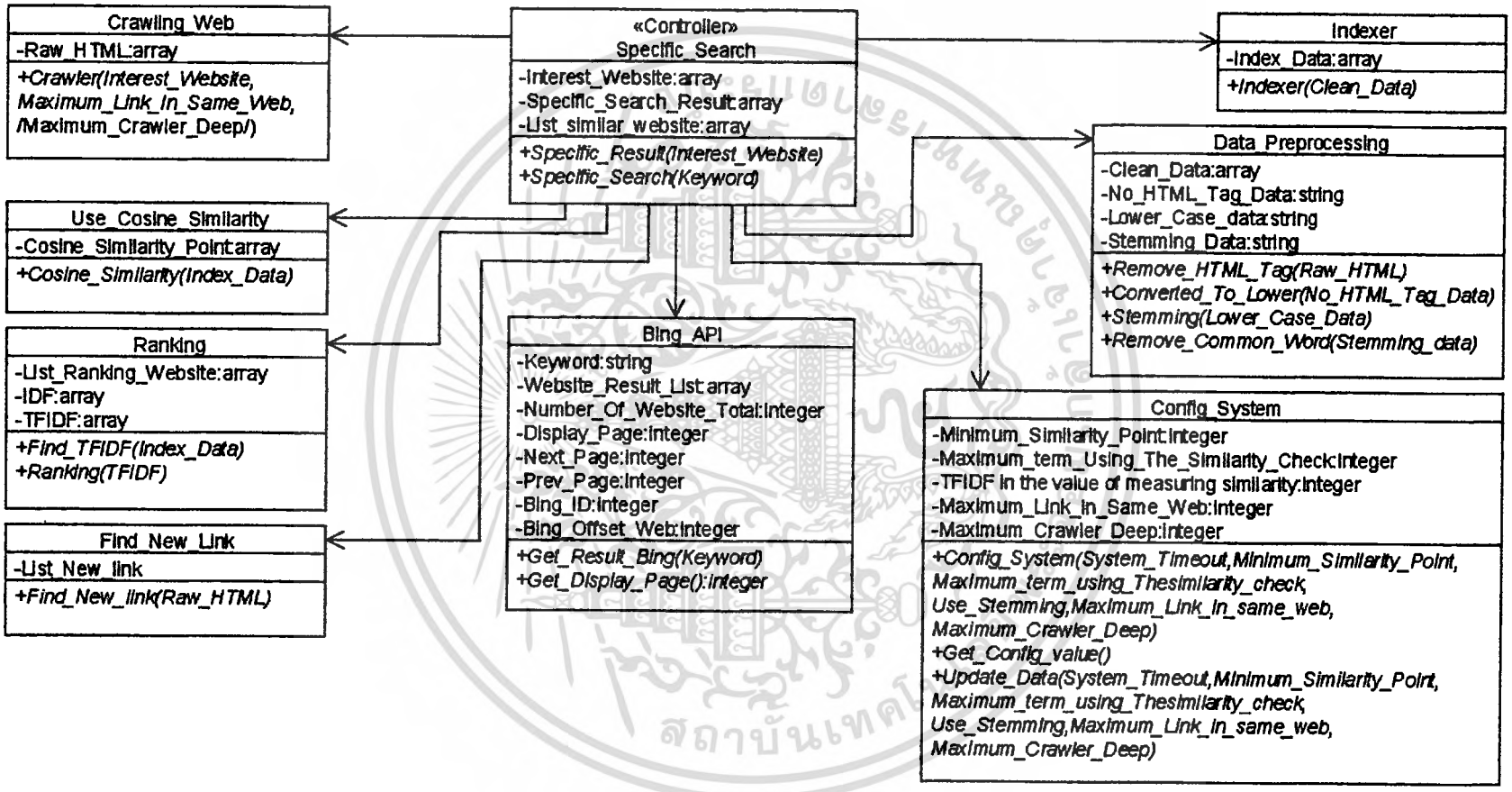
ยูสเคส	Find New Link
วัตถุประสงค์	เพื่อหาเว็บไซต์ในข้อมูลรูปแบบ HTML ที่ระบบส่งมา
เงื่อนไขเมื่อเริ่มต้น	ระบบข้อมูล HTML ข้อเว็บไซต์ที่ต้องการหาเว็บไซต์เพิ่มเติม
เมื่อทำงานสำเร็จ	ส่งลิงก์ที่ได้จากข้อมูล HTML ที่ได้รับให้กับระบบ
เมื่อทำงานไม่สำเร็จ	แจ้งว่าระบบไม่สามารถทำงานได้เพื่อให้ระบบส่งข้อมูล HTML มาให้อีกครั้ง
แอคเตอร์ที่เกี่ยวข้อง	System
สิ่งที่กระตุ้นการทำงาน	ระบบส่งเว็บไซต์ที่เลือกไว้ และชุดเว็บไซต์ที่หาเพิ่มเติมได้
อินพุต	เว็บไซต์ที่ระบบต้องการหาลิงก์เพิ่มเติมในรูปแบบข้อมูล HTML
เอาต์พุต	ลิงก์ที่อยู่ภายในข้อมูล HTML ของเว็บไซต์ที่ระบบส่งมา
รายละเอียด	1. ระบบส่งข้อมูลในรูปแบบ HTML ของเว็บไซต์ที่ระบบต้องการให้หาลิงก์ภายในนั้น 2. ส่งลิงก์ที่อยู่ภายในเว็บไซต์ที่ระบบส่งมาให้ โดยจำนวนมากสุดนั้นอยู่ที่ผู้ดูแลระบบได้กำหนดไว้



รูปที่ 4.17 แอ็กทิวิตีไดอะแกรมของระบบ Find_New_Link



รูปที่ 4.18 ซีเควนซ์ไดอะแกรมของระบบ Find_New_Link



รูปที่ 4.19 คลาสไดอะแกรมระบบค้นหาเว็บแบบเจาะจง

ตารางที่ 4.10 ซีอาร์ซี ของ Class Specific Search

Class Name: Specific_Search	
Class Type: Concrete class , Controller classes	
Attributes:	
Interest_Website	ค่าเว็บไซต์ที่ผู้ใช้สนใจ
Specific_Search_Result	ผลลัพธ์ในการค้นหา
List_similar_website	รายการเว็บไซต์ที่เหมือนกับเว็บไซต์ที่ผู้ใช้เลือก
Responsibility:	**Collaborator:**
Specific_Result(Interest_Website) หาเว็บไซต์ที่เหมือนกับเว็บไซต์ที่ผู้ใช้เลือกไว้	Use_Cosine_Similarity.Cosine_ Similarity(Cosine_Similarity(Index_Data) หาความเหมือนของเว็บไซต์คู่ใดคู่หนึ่ง Find_New_link .Find_New_link(Raw_HTML) หาลิงก์ภายในเว็บไซต์ Data_Preprocessing.Remove_HTML_Tag(Raw_HTML) เตรียมข้อมูลสำหรับการคำนวน Indexer.Indexer(Clean_Data) ทำสารบัญเว็บไซต์และเก็บไว้เพื่อให้สามารถเรียกใช้ได้ต่อไป โดยเก็บไว้ในรูปของ TF Crawling_Web .Crawler(Interest_website) ไปยังเว็บไซต์และดาวน์โหลดข้อมูลเว็บไซต์เก็บไว้ โดยเก็บข้อมูลเว็บไซต์ที่ผู้ใช้เลือกไว้ในรูปแบบ HTML
Specific_Search(Keyword) จัดการผลลัพธ์ที่ผู้ใช้เลือก โดยแสดงเว็บไซต์ที่มาจาก Bing API และ แสดงผลพร้อมให้ผู้ใช้เลือก	Bing_API.Get_Result_Bing(Keyword) เก็บผลลัพธ์ที่ได้จาก Bing API

ตารางที่ 4.11 ซีอาร์ซี ของ Class Crawling Web

Class Name: Crawling_Web	
Class Type: Concrete class , Entity classes	
Attributes:	
Raw_HTML	เก็บข้อมูลเว็บไซต์ที่ผู้ใช้เลือกไว้ในรูปแบบ HTML
Responsibility:	**Collaborator:**
Crawler(Interest_Website) ไปยังเว็บไซต์และดาวน์โหลดข้อมูลเว็บไซต์เก็บไว้ โดยเก็บข้อมูลเว็บไซต์ที่ผู้ใช้เลือกไว้ในรูปแบบ HTML	Use_Cosine_Similarity. Cosine_ Similarity(Cosine_Similarity(Index_Data) หาความเหมือนของเว็บไซต์คู่ใดคู่หนึ่ง Find_New_link .Find_New_link(Raw_HTML) หาลิงก์ภายในเว็บไซต์ Data_Preprocessing.Remove_HTML_Tag(Raw_HTML) เตรียมข้อมูลสำหรับการคำนวน Indexer.Indexer(Clean_Data) ทำสารบัญเว็บไซต์และเก็บไว้เพื่อให้สามารถเรียกใช้ได้ต่อไปโดยเก็บไว้ในรูปของ TF

ตารางที่ 4.12 ซีอาร์ซี ของ Class Use Cosine Similarity

Class Name: Use_Cosine_Similarity	
Class Type: Concrete class , Entity classes	
Attributes:	
Cosine_Similarity_Point	เก็บข้อมูลความเหมือนกันของเว็บไซต์คู่ใดคู่หนึ่ง
Responsibility:	**Collaborator:**
Cosine_Similarity(Index_Data) หาค่าความเหมือนกันของเว็บไซต์คู่ใดคู่หนึ่ง	

ตารางที่ 4.13 ซีอาร์ซี ของ Class Ranking

Class Name: Ranking	
Class Type: Concrete class , Entity classes	
Attributes:	
List_Ranking_Website	เก็บเว็บไซต์ที่เหมือนกับเว็บไซต์ที่ผู้ใช้เลือกไว้พร้อมเรียงลำดับค่าความเหมือน
IDF	เก็บค่า IDF ของคำทั้งหมด
TFIDF	เก็บค่า TFIDF ของคำในเว็บไซต์
Responsibility:	**Collaborator:**
Find_TFIDF(Index_Data) หาค่า TFIDF	Data_Preprocessing.Remove_HTML_Tag(Raw_HTML) เตรียมข้อมูลสำหรับการคำนวน Indexer.Indexer(Clean_Data) ทำสารบัญเว็บไซต์และเก็บไว้เพื่อให้สามารถเรียกใช้ได้ต่อไปโดยเก็บไว้ในรูปของ TF
Ranking(TFIDF) หาค่าความเหมือนกันของเว็บไซต์พร้อมทั้งเรียงลำดับ	

ตารางที่ 4.14 ซีอาร์ซี ของ Class Find New Link

Class Name: Find_New_Link	
Class Type: Concrete class , Entity classes	
Attributes:	
List_New_link	เว็บไซต์ใหม่ที่มาจากเว็บไซต์ที่ผู้ใช้เลือกไว้
Responsibility:	**Collaborator:**
Find_New_link(Raw_HTML) หาลิงก์ใหม่ที่อยู่ในเว็บไซต์ที่ส่งมา	

ตารางที่ 4.15 ซีอาร์ซี ของ Bing API

Class Name: Bing_API	
Class Type: Concrete class , Entity classes	
Attributes:	
Keyword	เว็บไซต์ใหม่ที่มาจากเว็บไซต์ที่ผู้ใช้เลือกไว้
Website_Result_List	เว็บไซต์ผลลัพธ์จาก Bing API
Number_Of_Website_Total	จำนวนเว็บไซต์ผลลัพธ์ทั้งหมด
Display_Page	หน้าที่แสดงผลลัพธ์อยู่
Next_Page	หน้าต่อไปที่จะแสดงผลลัพธ์
Prev_Page	หน้าก่อนหน้าที่แสดงผลลัพธ์
Bing_ID	รหัสของ Bing API
Bing_Offset_Web	จำนวนเว็บไซต์ที่แสดงผลต่อหนึ่งหน้า
Responsibility:	**Collaborator:**
Get_Result_Bing(Keyword) ส่งคำค้นไปยังระบบของ Bing Api และนำผลลัพธ์มาเก็บไว้	
Get_Display_Page(Display_page) ดึงหน้าที่แสดงผลลัพธ์ของ Bing API ที่แสดงอยู่	

ตารางที่ 4.16 ซีอาร์ซี ของ Class Indexer

Class Name: Indexer	
Class Type: Concrete class , Entity classes	
Attributes:	
Index_Data	ค่า TFIDF ของแต่ละคำในเว็บไซต์
Responsibility:	**Collaborator:**
Indexer(Clean_Data) นับความถี่ของคำเพื่อหาค่า TF	

ตารางที่ 4.17 ซีอาร์ซี ของ Class Data Preprocessing

Class Name: Data_Preprocessing	
Class Type: Concrete class , Entity classes	
Attributes:	
Clean_Data	คำในเว็บไซต์ที่ผ่านกระบวนการ Data Preprocessing แล้ว
No_HTML_Tag_Data	คำที่ผ่านกระบวนการกำจัด HTML Tag แล้ว
Lower_Case_data	คำที่ผ่านการแปลงให้เป็นตัวพิมพ์เล็กเรียบร้อยแล้ว
Stemming_Data	คำที่ผ่านกระบวนการ Stemming แล้ว
Responsibility:	**Collaborator:**
Remove_HTML_Tag(Raw_HTML) กำจัด HTML Tag ที่อยู่ในเอกสาร	
Converted_To_Lower(No_HTML_Tag_Data) แปลงตัวอักษรให้อยู่ในรูปแบบตัวพิมพ์เล็ก	
Stemming(Lower_Case_Data) ทำ Stemming	
Remove_Common_Word(Stemming_data) ลบคำที่ไม่มีความหมาย	

ตารางที่ 4.18 ซีอาร์ซี ของ Class Config System

Class Name: Config_System	
Class Type: Concrete class , Entity classes , Database Class	
Attributes:	
Minimum_Similarity_Point	ค่าความเหมือนต่ำสุดที่จะนำเว็บไซต์นั้นมาเป็นผลลัพธ์
Maximum_term_Using_The_Similarity_Check	ค่าคำสูงสุดที่จะนำมาใช้เพื่อเปรียบเทียบโดยวิธี Cosine Similarity
TFIDF in the value of measuring similarity	กำหนดว่าต้องการให้ใช้ค่า TF หรือ TFIDF มาใช้ในการหาค่าความเหมือน
Maximum_Link_In_Same_Web	จำนวนลิงก์สูงสุดที่จะให้เก็บจากหนึ่งเว็บไซต์
Maximum_Crawler_Deep	จำนวนชั้นมากสุดที่จะให้คำลงไป
Responsibility:	**Collaborator:**
Config_System(System_Timeout,Minimum_Similarity_Point, Maximum_term_using_Thesimilarity_check, Use_Stemming,Maximum_Link_in_same_web, Maximum_Crawler_Deep) รับค่าใหม่	
Get_Config_value() เก็บค่าเดิมที่ตั้งไว้	
Update_Data(System_Timeout,Minimum_Similarity_Point, Maximum_term_using_Thesimilarity_check, Use_Stemming,Maximum_Link_in_same_web, Maximum_Crawler_Deep) ปรับปรุงค่าใหม่	

บทที่ 5
การออกแบบส่วนติดต่อกับผู้ใช้

5.1 รายละเอียดซอฟต์แวร์ที่ใช้ในการพัฒนาระบบ

- โปรแกรมที่ใช้ในการพัฒนาระบบคือ NetBeans IDE 6.8

- ใช้ภาษา Python โดยส่วนที่เป็น Web Application ใช้ Django Web framework เป็นตัวควบคุมการแสดงผลผ่าน HTML และทำหน้าที่เป็น Server

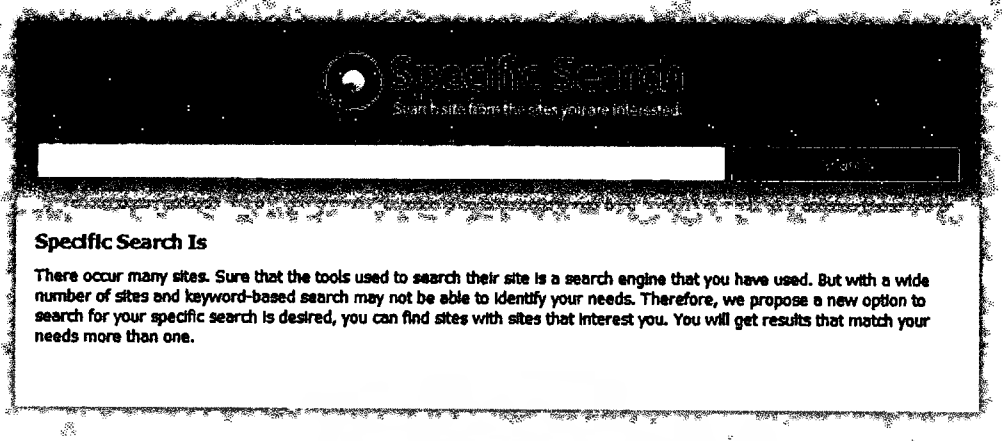
5.2 การออกแบบโครงร่างของระบบ

เนื่องจากการใช้งาน Django Framework ซึ่งมีสถาปัตยกรรมในรูปแบบของ Model–View–Controller (MVC) ทำให้การออกแบบระบบและการออกแบบในส่วนติดต่อกับผู้ใช้สามารถแยกออกจากกัน และพัฒนาไปพร้อมกันได้ การออกแบบในส่วนติดต่อกับผู้ใช้งานสามารถแบ่งออกเป็น 2 ส่วนด้วยการคือส่วนสำหรับผู้ใช้งานระบบทั่วไป และส่วนสำหรับให้ผู้ดูแลระบบเข้าไปควบคุมจัดการปรับแก้ค่าต่าง

5.2.1 ส่วนติดต่อกับผู้ใช้งานทั่วไป

1. หน้าแรกของเว็บไซต์

หน้าแรกของเว็บไซต์จะเป็นส่วนที่แนะนำการใช้งานระบบ โดยจะมีช่องให้ผู้ใช้งานสามารถกรอกคำที่ใช้ในการค้นหาเว็บไซต์ที่ต้องการ โดยด้านล่างจะเป็นคำบรรยายว่าระบบคืออะไร ทำงานอย่างไร เพื่อให้ผู้ที่ใช้งานในครั้งแรกสามารถทำความเข้าใจระบบได้ และสามารถใช้งานระบบในส่วนอื่นๆได้ ดังแสดงในรูปที่ 5.1



รูปที่ 5.1 หน้าแรกของระบบค้นหาแบบเจาะจง

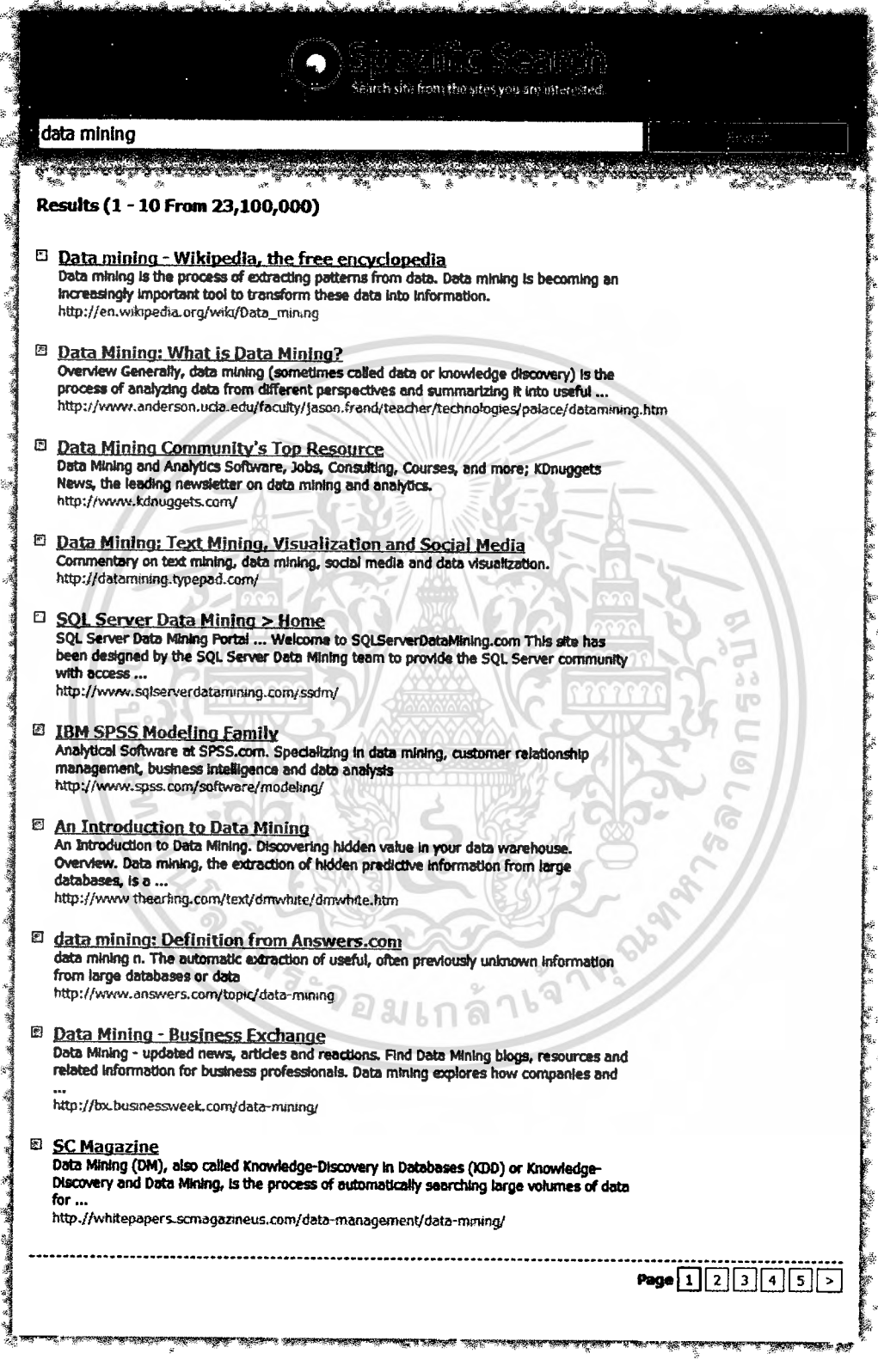
2. หน้าแสดงผลเว็บไซต์ที่ได้จากคำที่ผู้ใช้ใช้ในการค้นหา

หน้านี้จะเป็นหน้าผลลัพธ์ที่ได้จาก Bing API แสดงเว็บไซต์ที่เกี่ยวข้องกับคำที่ผู้ใช้ใช้ในการค้นหา โดยจะแสดงเว็บไซต์จำนวน 10 เว็บไซต์ต่อหน้า ด้านบนสุดจะแสดงจำนวนผลลัพธ์ทั้งหมด ในส่วนด้านล่างจะเป็นลิงก์เพื่อแสดงผลหน้าอื่นๆต่อไป

ในทุกผลลัพธ์จะมีเช็คบ๊อก อยู่ให้ผู้ใช้งานสามารถเลือกเว็บไซต์โดยการเลือกที่เช็คบ๊อก จากนั้น ระบบจะทำการเลือกผลลัพธ์ที่ผู้ใช้เลือกเป็นเว็บไซต์ที่ผู้ใช้งานสนใจ และจะนำผู้ใช้ไปสู่หน้าค้นหา โดยใช้เว็บไซต์ที่ผู้ใช้สนใจโดยอัตโนมัติ

หากผู้ใช้งานต้องการค้นหาด้วยคำค้นอื่นๆ ผู้ใช้งานก็สามารถกรอกคำค้นไปที่ช่องด้านบน และคลิกที่ search ได้ทันทีหากผู้ใช้งานคลิกที่ลิงก์ผลลัพธ์ ระบบจะแสดงหน้าตัวอย่างเว็บไซต์

หน้าแสดงผลเว็บไซต์ที่ได้จากคำที่ผู้ใช้ใช้ในการค้นหาแสดงดังรูปที่ 5.1



รูปที่ 5.2 หน้าผลลัพธ์ที่ได้จากการค้นหาด้วยคำค้น

3. หน้าค้นหาโดยใช้เว็บไซต์ที่ผู้ใช้สนใจ

หน้านี้เป็นหน้าที่เรียกใช้งานระบบการค้นหาแบบเจาะจง โดยหลังจากที่ผู้ใช้งานได้เลือกเว็บไซต์ที่สนใจจากหน้าแสดงผลลัพธ์ที่ได้จากการค้นหาด้วยคำค้น โดยการเลือกที่เช็คบ๊อกระบบจะทำการนำผู้ใช้มาที่หน้านี้โดยอัตโนมัติ

การเลือกเว็บไซต์ที่สนใจเพิ่มเติมก็ให้เลือกที่เช็คบ๊อก ของผลลัพธ์ด้านล่างเพิ่ม แต่ถ้าหากต้องการลบเว็บไซต์ที่สนใจให้คลิกที่เช็คบ๊อกที่ได้เลือกไว้จากนั้นระบบจะทำการลบเว็บไซต์ที่เลือกโดยอัตโนมัติ ดังรูปที่ 5.3

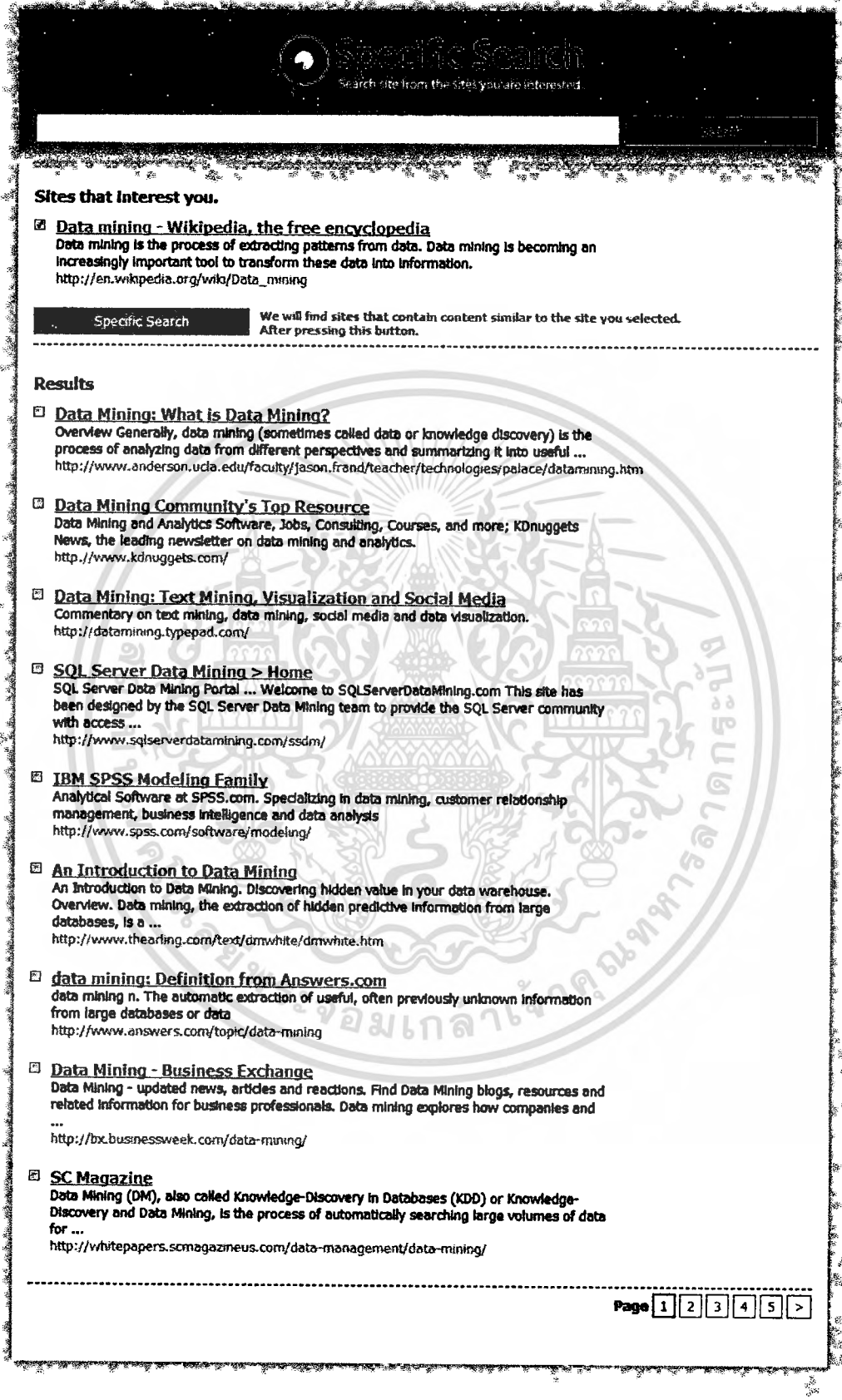
เมื่อผู้ใช้ได้เว็บไซต์ที่สนใจมาจำนวนหนึ่ง หรือเป็นที่พอใจแล้วก็ให้ผู้ใช้คลิกที่ปุ่ม Specific Search จากนั้นระบบจะทำการนำเว็บไซต์ที่ผู้ใช้ได้เลือกไว้ไปประมวลผลต่อไป เพื่อหาเว็บไซต์ที่มีความเหมือนกับเว็บไซต์ที่ผู้ใช้งานได้เลือกไว้ และแสดงในหน้าผลลัพธ์การค้นหาแบบเจาะจง

4. หน้าแสดงตัวอย่างเว็บไซต์

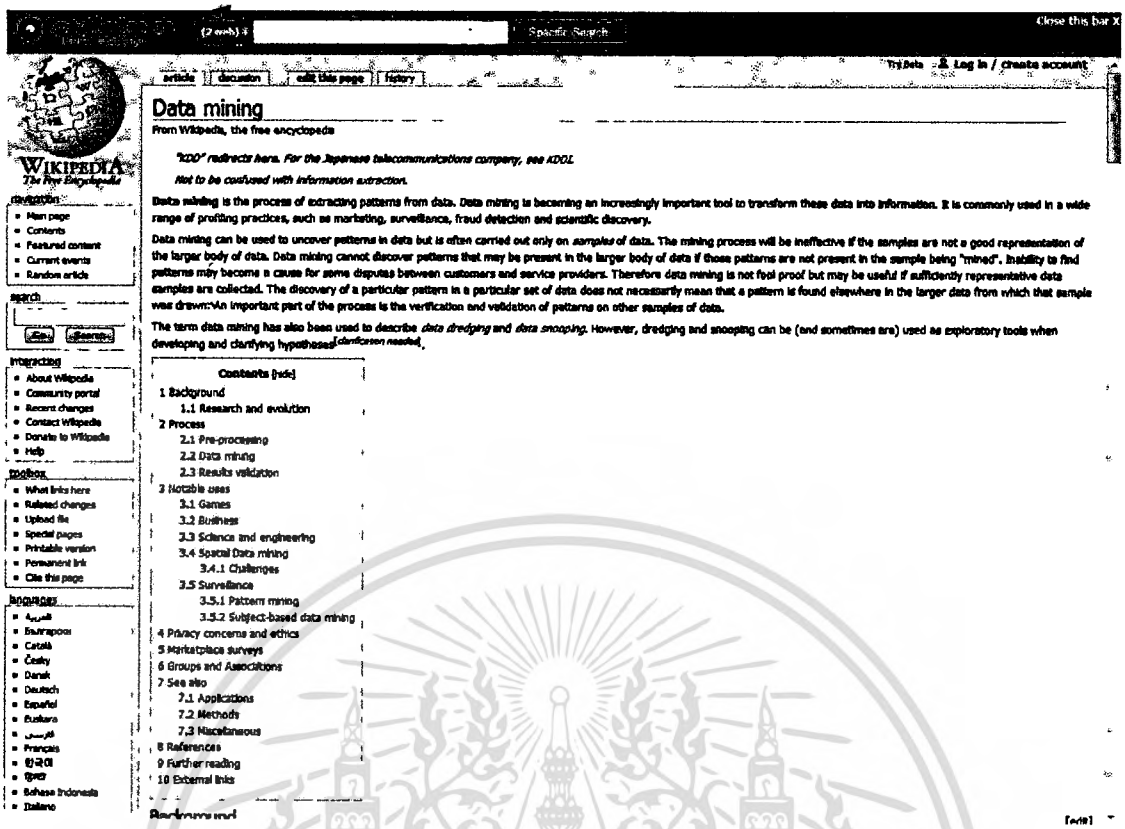
หากผู้ใช้งานต้องการดูเว็บไซต์ สามารถดูได้โดยคลิกที่ลิงก์สีฟ้า ระบบจะแสดงดังรูปที่ 5.4 โดยแสดงเนื้อหาเว็บไซต์ด้านล่าง และมีรายละเอียดเป็นแถบสีฟ้าด้านบนให้ผู้ใช้งานสามารถกลับไปยังหน้าแสดงลิงก์ทั้งหมดได้โดยคลิกที่โลโก้ Specific Search หรือเลือกที่จะค้นหาใหม่ได้ หากผู้ใช้สนใจที่จะใช้การค้นหาแบบเจาะจงก็สามารถคลิกได้ที่ปุ่ม Specific Search ได้ทันที นอกจากนี้ หากผู้ใช้ต้องการเข้าชมเว็บไซต์นี้โดยไม่ต้องการที่จะใช้บริการ การค้นหาแบบเจาะจงก็สามารถคลิกที่ปุ่ม Close this bar เพื่อเข้าชมเว็บไซต์แบบปรกติได้ทันที

5. หน้าแสดงผลลัพธ์การค้นหาแบบเจาะจง

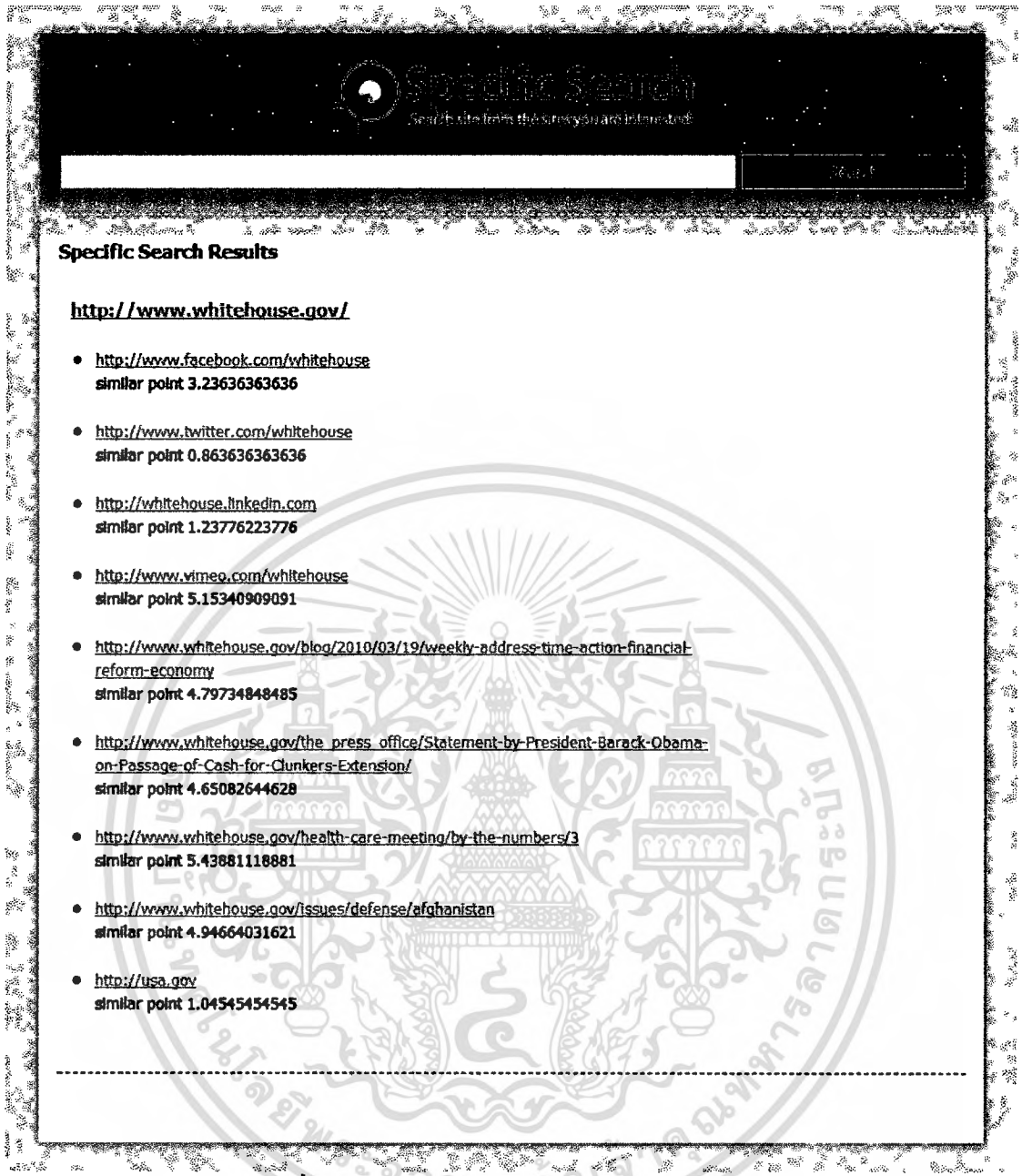
หน้านี้แสดงผลลัพธ์ที่ได้จากการค้นหาแบบเจาะจง โดยจะแสดงเว็บไซต์ที่ผู้ใช้ได้เลือกไว้ก่อน จากนั้นในส่วนด้านล่างจะแสดงเว็บไซต์ที่ได้จากการค้นหาโดยผ่านระบบการค้นหาแบบเจาะจง โดยในแต่ละเว็บไซต์จะมีค่าความเหมือนกับเว็บไซต์ที่ผู้ใช้ได้เลือกไว้ก่อนอยู่ด้านล่างสุดสีส้ม



รูปที่ 5.3 หน้าผลลัพธ์ที่ได้จากการค้นหาด้วยคำค้น



รูปที่ 5.4 หน้าแสดงตัวอย่างเว็บไซต์



รูปที่ 5.5 หน้าแสดงผลลัพธ์การค้นหาแบบเจาะจง

5.2.2 ส่วนติดต่อกับผู้ดูแลระบบ

1. หน้าจอล็อคอินเข้าสู่ระบบ

เป็นหน้าจอสำหรับผู้ดูแลระบบ ใช้สำหรับเข้าไปปรับตั้งค่าระบบการค้นหาแบบเจาะจง โดยผู้ใช้จะต้องกรอกรหัสผ่าน และชื่อผู้ใช้ก่อนจึงจะสามารถเข้าใช้งานระบบได้ หากชื่อผู้ใช้หรือรหัสผ่านผิดพลาดระบบจะปฏิเสธการร้องขอดังกล่าว พร้อมแจ้งข้อผิดพลาดให้ทราบ โดยหน้าจอสำหรับล็อคอินเข้าสู่ระบบแสดงดังรูปที่ 5.6

2. หน้าจอปรับตั้งค่าระบบ

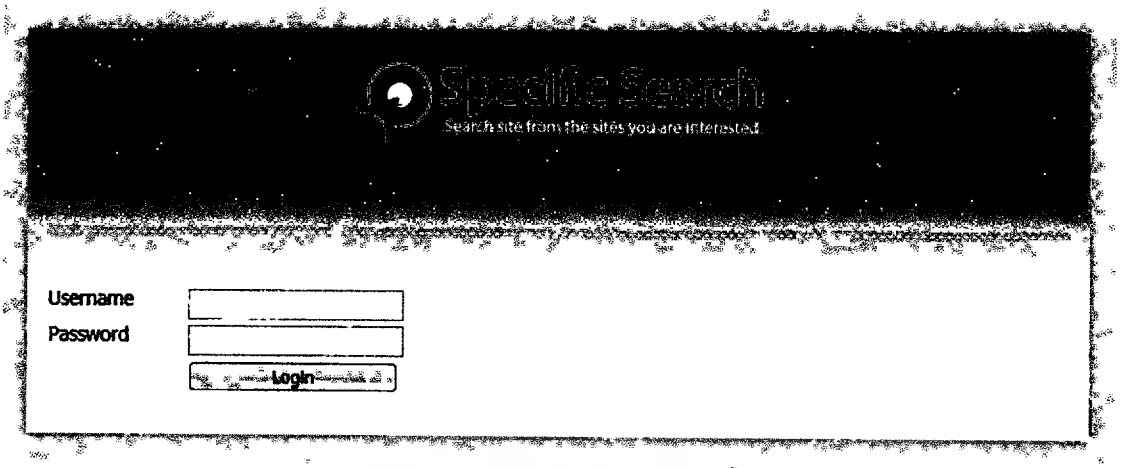
หลังจากที่ผู้ใช้งานได้ล็อคอินเข้าสู่ระบบเรียบร้อยแล้ว จะปรากฏหน้าแสดงดังรูปที่ 5.7 โดยจะแสดงค่ามาตราฐานการตั้งค่าสำหรับการใช้งานระบบการค้นหาแบบเจาะจงโดยมีรายละเอียดดังนี้

ตารางที่ 5.1 แสดงรายละเอียดค่าที่ใช้ในการปรับระบบการค้นหาแบบเจาะจง

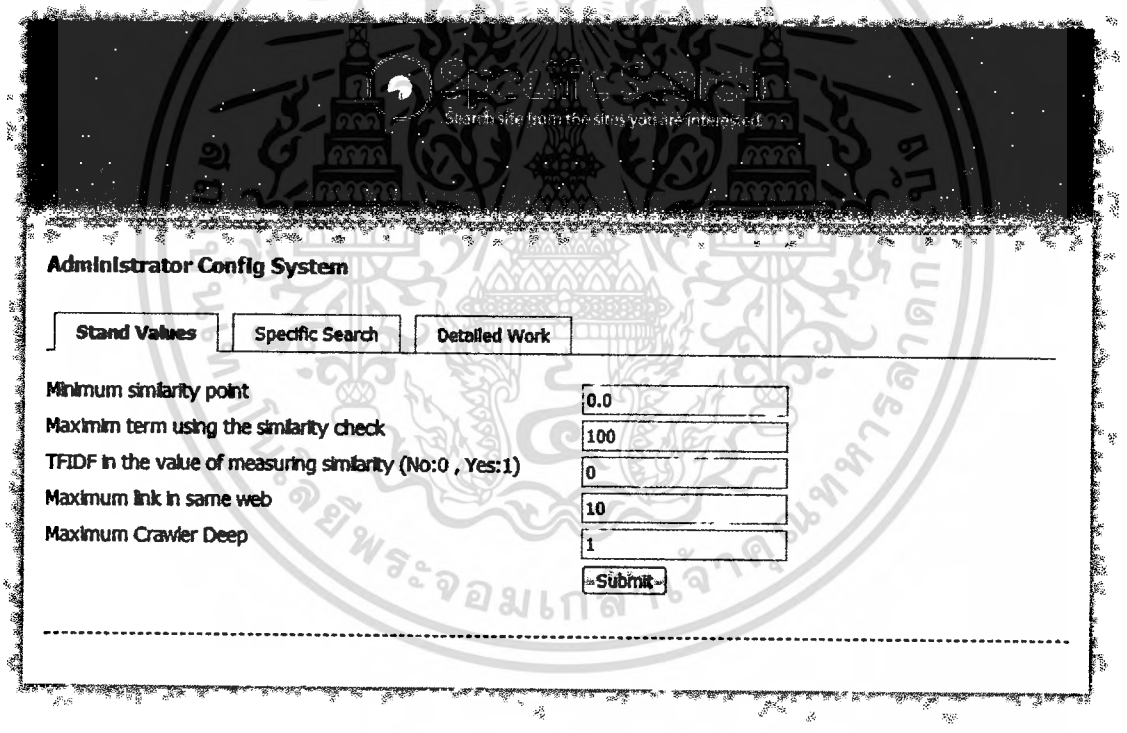
ชื่อเมนู	คำอธิบาย
Standard Value	ค่ามาตราฐานสำหรับตั้งค่าการค้นหาแบบเจาะจง โดยมีรายละเอียดแสดงดังตารางที่ 5.2
Specific Search	เป็นผลการค้นหาแบบเจาะจง โดยมีการทำงานเหมือนกับที่ผู้ใช้งานทั่วไปใช้
Detail Work	รายละเอียดการทำงานของระบบ มีไว้สำหรับให้ผู้ดูแลระบบทราบรายละเอียดการทำงานของระบบและสามารถปรับแก้ไขการทำงานได้

ตารางที่ 5.2 แสดงรายละเอียดค่าที่ใช้ในการปรับระบบการค้นหาแบบเจาะจงในส่วน Standard Value

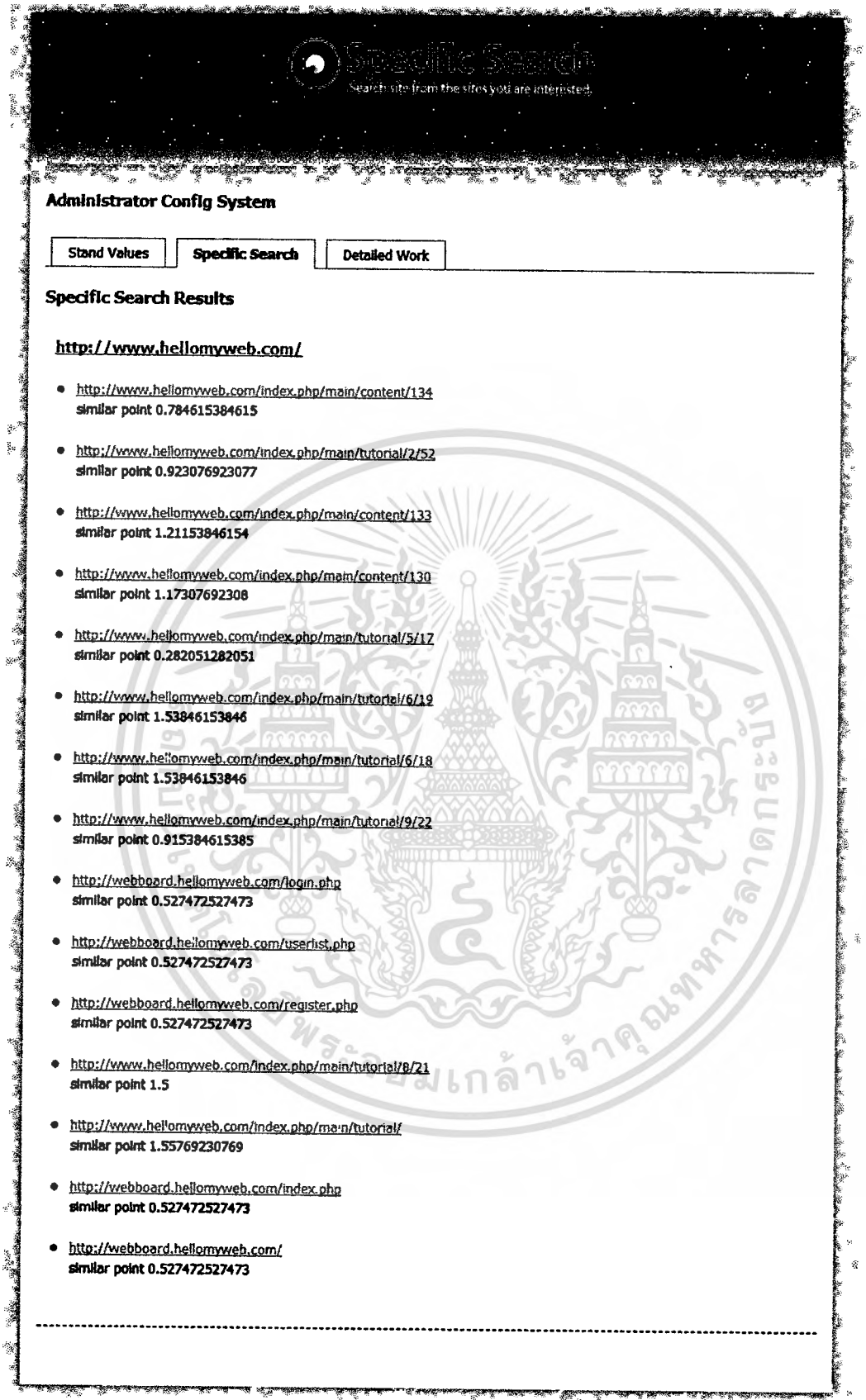
ชื่อเมนู	คำอธิบาย
Minimum Similarity Point	ค่าความเหมือนที่น้อยที่สุดที่จะอนุญาติให้มีการเก็บเว็บไซต์นี้ไว้พิจราณาต่อ
Maximum term using the similarity check	ค่าจำนวนคำมากที่สุดที่จะนำไปใช้ในการคิดคำนวนค่าความเหมือนกันของเว็บไซต์ 2 เว็บ ด้วยวิธี Cosine Similarity
TFIDF in the value of measuring similarity	การตั้งค่าให้มีการนำค่า TFIDF มาใช้งานในการคำนวนด้วยวิธี Cosine Similarity หรือไม่
Maximum link in same web	จำนวนลิงก์มากสุดที่ให้หาจากในเว็บไซต์เดียวกัน
Maximum Crawler Deep	จำนวนความลึกจากเว็บไซต์ตั้งต้น



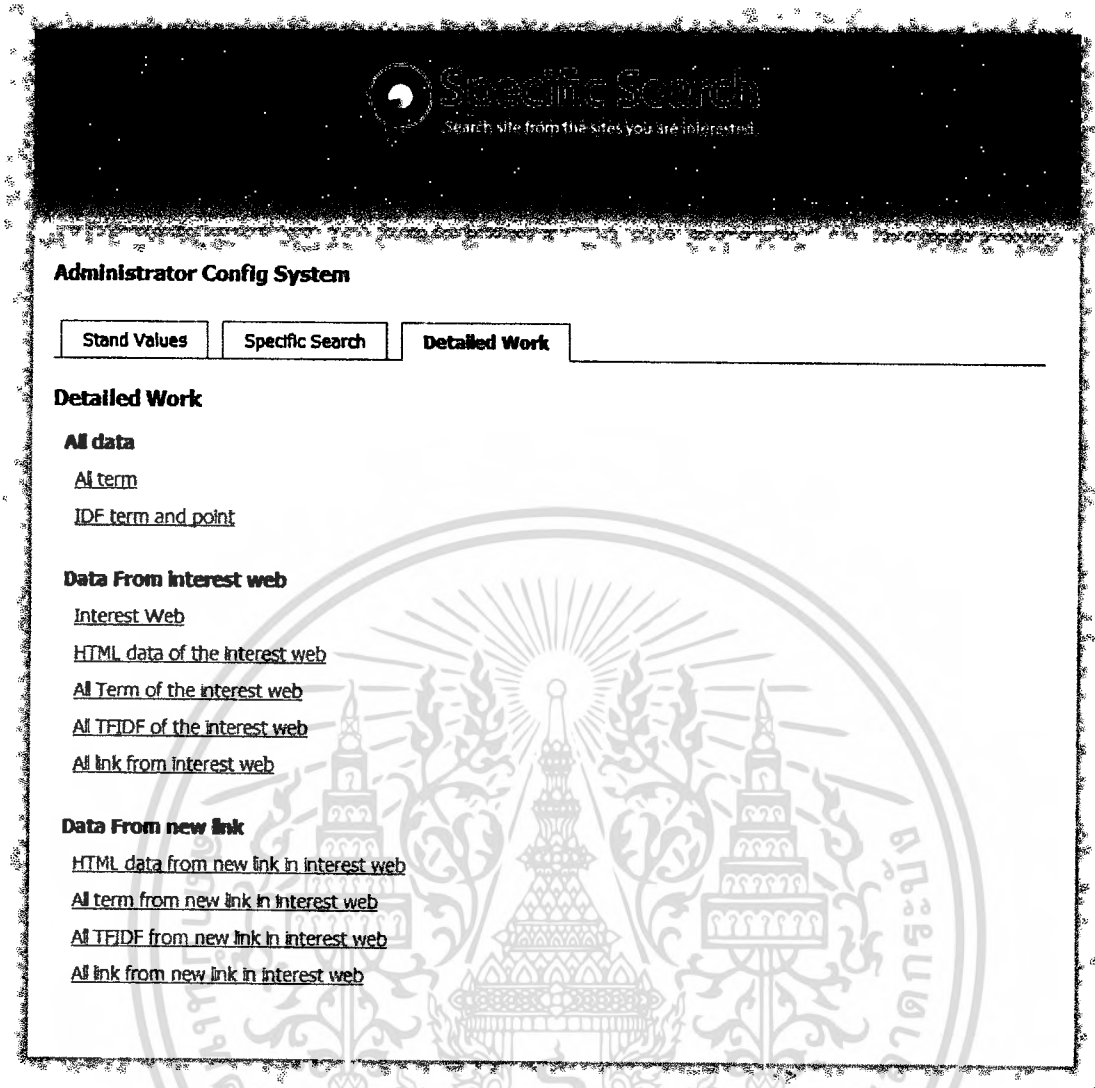
รูปที่ 5.6 หน้าจอล็อคอินเข้าสู่ระบบ



รูปที่ 5.7 หน้าจอปรับตั้งค่าระบบ



รูปที่ 5.8 หน้าจอแสดงผลลัพธ์สำหรับผู้ดูแลระบบ



รูปที่ 5.9 หน้าจอแสดงผลลัพธ์การคำนวนสำหรับผู้ดูแลระบบ

3. หน้าจอแสดงผลลัพธ์สำหรับผู้ดูแลระบบ

เป็นการแสดงผลลัพธ์เหมือนกับของผู้งานปรกติ นำมาแสดงไว้ในส่วนนี้เพื่อให้ผู้ดูแลระบบสามารถปรับค่าต่างๆ พร้อมดูผลลัพธ์ได้ง่ายยิ่งขึ้นโดยแสดงดังรูปที่ 5.8

4. หน้าจอแสดงผลลัพธ์การคำนวนสำหรับผู้ดูแลระบบ

สำหรับผู้ดูแลระบบค่าในส่วนนี้จะเป็นค่าสำหรับไว้ตั้งค่าระบบให้ทำงานได้อย่างมีประสิทธิภาพ โดยจะมีค่าคำที่ปรากฏมากที่สุดที่นำมาใช้ในการคำนวน และค่าลิงก์ที่เป็นผลลัพธ์พร้อมกับค่าที่คำนวนได้แสดงดังรูปที่ 5.9

รูปที่ 5.10 หน้าจอแสดงเทอมทั้งหมดที่ใช้ในการคำนวน

บทที่ 6
ผลการทดสอบการทำงานของระบบการค้นหาเว็บไซต์แบบเจาะจง

6.1 ผลการทดสอบการทำงานของระบบ Specific Web Search โดยเปรียบเทียบระหว่างการจัดอันดับด้วย Cosine Similarity และ K-NN

วิธีการในการจัดอันดับเว็บไซต์ว่ามีความเหมือนกับเว็บไซต์ที่ผู้ใช้เลือกมากน้อยเพียงใดนั้น สามารถทำได้หลายวิธีด้วยกัน แต่ในรายงานฉบับนี้จะเสนอวิธีการเปรียบเทียบความเหมือน 2 วิธี คือ

1. Cosine Similarity
2. K-NN

โดยทั้งสองวิธีมีขั้นตอนในการเปรียบเทียบต่างกับและค่าที่ได้จะมีความต่างกัน ค่า Cosine Similarity นั้นจะใช้วิธีการนำค่า TFIDF ของคำมาคอทกัน ทำให้ค่าที่ได้หากมีค่ามากแสดงว่ามีความเหมือนมาก ส่วน K-NN นั้นจะทำค่า TFIDF มาลบกัน ทำให้หากได้ค่าน้อยแสดงว่าเอกสารทั้งสองมีความเหมือนกันมาก

6.1.1 การตั้งค่าสำหรับการจัดอันดับด้วยวิธี Cosine Similarity

$$sim(q,p) = \frac{\sum_{k \in q \cap p} f_{kq} f_{kp}}{\sqrt{\left(\sum_{k \in p} f_{kp}^2\right)\left(\sum_{k \in q} f_{kp}^2\right)}}$$

โดย

จำนวนคำที่ใช้ในการคำนวนคือ 100 คำ

ใช้ค่า TFIDF ในการคำนวน

จำนวนลิงก์มากสุดที่ให้หาจากในเว็บไซต์เดียวกันคือ 10

จำนวนความลึกจากเว็บไซต์ตั้งต้นคือ 1

ค่า Cosine Similarity ต่ำสุดที่นำมาแสดงคือ 0.2

6.1.2 การตั้งค่าสำหรับการจัดอันดับด้วยวิธี K-NN

$$D(d1,d2) = \sqrt{\sum_{t=1}^{n}(X(t,d1) - X(t,d2))^2}$$

โดย

จำนวนคำที่ใช้ในการคำนวนคือ 100 คำ

ใช้ค่า TFIDF ในการคำนวน

จำนวนลิงก์มากสุดที่ให้หาจากในเว็บไซต์เดียวกันคือ 10

จำนวนความลึกจากเว็บไซต์ตั้งต้นคือ 1

ค่า K ที่นำมาใช้คือ 10

6.2 ผลการทดสอบ

6.2 .1 การทดลองด้วยคำค้นคำว่า Obama

เว็บไซต์ที่เลือกเป็นเว็บไซต์ตั้งต้นคือ http://www.whitehouse.gov/

ผลลัพธ์การเรียงลำดับด้วยวิธี Cosine Similarity

- http://whitehouse.linkedin.com
  similar point 0.989325999513
- http://www.whitehouse.gov/health-care-meeting/by-the-numbers/3
  similar point 0.877024993835
- http://www.whitehouse.gov/blog/2010/03/19/weekly-address-time-action-financial-reform-economy
  similar point 0.847198988687
- http://www.whitehouse.gov/the_press_office/Statement-by-President-Barack-Obama-on-Passage-of-Cash-for-Clunkers-Extension/
  similar point 0.822791072111
- http://www.whitehouse.gov/issues/defense/afghanistan
  similar point 0.800788688436
- http://www.vimeo.com/whitehouse
  similar point 0.753661100904
- http://www.facebook.com/whitehouse
  similar point 0.65092361792

- http://usa.gov

  similar point 0.6899818776

- http://www.twitter.com/whitehouse

  similar point 0.405014717284

ผลลัพธ์การเรียงลำดับด้วยวิธี K-NN

- http://whitehouse.linkedin.com

  similar point 0.0850198415456

- http://usa.gov

  similar point 0.154832387109

- http://www.whitehouse.gov/issues/defense/afghanistan

  similar point 0.274957905855

- http://www.whitehouse.gov/health-care-meeting/by-the-numbers/3

  similar point 0.286944001755

- http://www.whitehouse.gov/blog/2010/03/19/weekly-address-time-action-financial-reform-economy

  similar point 0.313538357895

- http://www.whitehouse.gov/the_press_office/Statement-by-President-Barack-Obama-on-Passage-of-Cash-for-Clunkers-Extension/

  similar point 0.369904066863

- http://www.facebook.com/whitehouse

  similar point 0.448185576642

- http://www.twitter.com/whitehouse

  similar point 0.452818670411

6.2 .2 การทดลองด้วยคำค้นคำว่า michael jackson

เว็บไซต์ที่เลือกเป็นเว็บไซต์ตั้งต้นคือ http://en.wikipedia.org/wiki/Michael_Jackson

ผลลัพธ์การเรียงลำดับด้วยวิธี Cosine Similarity

- http://mt.wikipedia.org/wiki/Michael_Jackson
  similar point 0.928244356269
- http://tpi.wikipedia.org/wiki/Michael_Jackson
  similar point 0.900804246597
- http://sl.wikipedia.org/wiki/Michael_Jackson
  similar point 0.810958123511
- http://www.allmusicguide.com/cg/amg.dll?p=amg&sql=10:rz60tr7qklkx
  similar point 0.780522615107
- http://www.timesonline.co.uk/tol/news/world/us_and_americas/article6808546.ece
  similar point 0.715329318765
- http://news.bbc.co.uk/onthisday/hi/dates/stories/august/24/newsid_2512000/2512077
  .stm
  similar point 0.672559813367
- http://www.presidency.ucsb.edu/ws/index.php?pid=18331
  similar point 0.542902256523

ผลลัพธ์การเรียงลำดับด้วยวิธี K-NN

- http://ilo.wikipedia.org/wiki/Michael_Jackson
  similar point 0.0588853989222
- http://mt.wikipedia.org/wiki/Michael_Jackson
  similar point 0.0776736685684
- http://tpi.wikipedia.org/wiki/Michael_Jackson
  similar point 0.103772277011
- http://sl.wikipedia.org/wiki/Michael_Jackson
  similar point 0.119964385611
- http://news.bbc.co.uk/onthisday/hi/dates/stories/august/24/newsid_2512000/2512077
  .stm
  similar point 0.337103139093
- http://www.allmusicguide.com/cg/amg.dll?p=amg&sql=10:rz60tr7qklkx

similar point 0.345811684143

6.2 .3 การทดลองด้วยคำค้นคำว่า Michael_Jordan

เว็บไซต์ที่เลือกเป็นเว็บไซต์ตั้งต้นคือ http://en.wikipedia.org/wiki/Michael_Jordan

ผลลัพธ์การเรียงลำดับด้วยวิธี Cosine Similarity

- http://en.wikipedia.org/w/index.php?title=Template:Sporting_News_College_Men%27s_Basketball_Player_of_the_Year&action=edit
  similar point 0.920188120485
- http://findarticles.com/p/articles/mi_m1355/is_n17_v92/ai_19783684
  similar point 0.880506503739
- http://www.nba.com/jordan/mj9091.html
  similar point 0.809733531474
- http://wikimediafoundation.org/
  similar point 0.764699997447
- http://www.nba.com/playerfile/michael_jordan/bio.html
  similar point 0.639455161757

ผลลัพธ์การเรียงลำดับด้วยวิธี K-NN

- http://en.wikipedia.org/w/index.php?title=Template:Sporting_News_College_Men%27s_Basketball_Player_of_the_Year&action=edit
  similar point 0.0911340694602
- http://en.wikipedia.org/w/index.php?title=Template:NBA_Rookies_of_the_Year&action=edit
  similar point 0.0943325213283
- http://wikimediafoundation.org/
  similar point 0.12293773571
- http://www.hoophall.com/hall-of-famers/tag/michael-jordan
  similar point 0.327921218873
- http://www.nba.com/jordan/mj9091.html
  similar point 0.340895858733

6.2 .4 การทดลองด้วยคำค้นคำว่า steve job

เว็บไซต์ที่เลือกเป็นเว็บไซต์ตั้งต้นคือ http://en.wikipedia.org/wiki/Steve_jobs

ผลลัพธ์การเรียงลำดับด้วยวิธี Cosine Similarity

- http://news.scotsman.com/comment/Steve-Jobs-profile-Apple39s-hard.4863847.jp
  similar point 0.882709108332
- http://www.dnaindia.com/money/report_what-steve-jobs-did-when-he-was-fired-from-apple_1254757
  similar point 0.846346283287
- http://en.wikipedia.org/w/index.php?title=Template:HP&action=edit
  similar point 0.401550144221

ผลลัพธ์การเรียงลำดับด้วยวิธี K-NN

- http://en.wikipedia.org/w/index.php?title=Template:HP&action=edit
  similar point 0.0972744135578
- http://gl.wikipedia.org/wiki/Steven_Paul_Jobs
  similar point 0.104421168545
- http://cs.wikipedia.org/wiki/Steve_Jobs
  similar point 0.100933671165
- http://news.scotsman.com/comment/Steve-Jobs-profile-Apple39s-hard.4863847.jp
  similar point 0.209236096947
- http://www.dnaindia.com/money/report_what-steve-jobs-did-when-he-was-fired-from-apple_1254757
  similar point 0.241077513316

6.2 .5 การทดลองด้วยคำค้นคำว่า Johnny Depp

เว็บไซต์ที่เลือกเป็นเว็บไซต์ตั้งต้นคือ http://www.imdb.com/name/nm*0000136*

ผลลัพธ์การเรียงลำดับด้วยวิธี Cosine Similarity

- http://www.imdb.pt
  similar point 0.460522962006
- http://www.imdb.de
  similar point 0.293251569746

- http://www.twitter.com/imdb

  similar point 0.201080844045

ผลลัพธ์การเรียงลำดับด้วยวิธี K-NN

- http://resume.imdb.com/

  similar point 0.161861613515

- http://resume.imdb.com/?name=Johnny%20Depp

  similar point 0.161861613515

- http://resume.imdb.com

  similar point 0.161861613515

- http://www.imdb.de

  similar point 0.251272444481

- http://www.imdb.pt

  similar point 0.420627757481

- http://www.twitter.com/imdb

  similar point 0.451721407949

- http://www.imdb.es

  similar point 0.472237276216

6.2 .6 การทดลองด้วยคำค้นคำว่า Will Smith

เว็บไซต์ที่เลือกเป็นเว็บไซต์ตั้งต้นคือ http://en.wikipedia.org/wiki/Will_Smith

ผลลัพธ์การเรียงลำดับด้วยวิธี Cosine Similarity

- http://en.wikinews.org/wiki/Scientology_ties_at_New_Village_Leadership_Academy_stir_controversy_for_Will_Smith_and_Jada_Pinkett-Smith

  similar point 0.407093989586

- http://id.wikipedia.org/wiki/Will_Smith

  similar point 0.737045799691

-

- http://uk.wikipedia.org/wiki/%D0%92%D1%96%D0%BB%D0%BB_%D0%A1%D0%BC%D1%96%D1%82

similar point 0.862192780006

- http://ja.wikipedia.org/wiki/%E3%82%A6%E3%82%A3%E3%83%AB%E3%83%BB%E3%82%B9%E3%83%9F%E3%82%B9
  similar point 0.881624692523
- http://hr.wikipedia.org/wiki/Will_Smith
  similar point 0.894163659189
- http://ckb.wikipedia.org/wiki/%D9%88%DB%8C%DA%B5_%D8%B3%D9%85%DB%8C%D8%AA
  similar point 1.0

ผลลัพธ์การเรียงลำดับด้วยวิธี K-NN

- http://ckb.wikipedia.org/wiki/%D9%88%DB%8C%DA%B5_%D8%B3%D9%85%DB%8C%D8%AA
  similar point 0.028341420667
- http://ta.wikipedia.org/wiki/%E0%AE%B5%E0%AE%BF%E0%AE%B2%E0%AF%8D_%E0%AE%9A%E0%AE%BF%E0%AE%AE%E0%AE%BF%E0%AE%A4%E0%AF%8D
  similar point 0.05805434857
- http://ja.wikipedia.org/wiki/%E3%82%A6%E3%82%A3%E3%83%AB%E3%83%BB%E3%82%B9%E3%83%9F%E3%82%B9
  similar point 0.115835349321
- http://uk.wikipedia.org/wiki/%D0%92%D1%96%D0%BB%D0%BB_%D0%A1%D0%BC%D1%96%D1%82
  similar point 0.143150295359
- http://de.wikipedia.org/wiki/Will_Smith
  similar point 0.174501470545
- http://id.wikipedia.org/wiki/Will_Smith
  similar point 0.267306190089
- http://www.foxnews.com/story/0,2933,316808,00.html
  similar point 0.314313587006
- http://jam.canoe.ca/Movies/Artists/S/Smith_Will/2008/03/23/5078376-sun.html
  similar point 0.489506793723

- http://en.wikinews.org/wiki/Scientology_ties_at_New_Village_Leadership_Academy_stir_controversy_for_Will_Smith_and_Jada_Pinkett-Smith
  similar point 0.566392983062
- http://hr.wikipedia.org/wiki/Will_Smith
  similar point 0.596597534299

6.2 .7 การทดลองด้วยคำค้นคำว่า Tom Cruise

เว็บไซต์ที่เลือกเป็นเว็บไซต์ตั้งต้นคือ *http://www.imdb.com/name/nm*0000129/

ผลลัพธ์การเรียงลำดับด้วยวิธี Cosine Similarity

- http://pro.imdb.com/rg/maindetails-title/nconst-pro-header-link/name/nm0000129/
  similar point 0.573759334618
- http://www.imdb.es
  similar point 0.56630391076
- http://jam.canoe.ca/Movies/Artists/S/Smith_Will/2008/03/23/5078376-sun.html
  similar point 0.403729066577
- http://www.amazon.com/exec/obidos/redirect-home/internetmoviedat
  similar point 0.365571970535
- http://www.imdb.de
  similar point 0.238884790246
- http://resume.imdb.com/
  similar point 0.148331999404

ผลลัพธ์การเรียงลำดับด้วยวิธี K-NN

- http://resume.imdb.com/
  similar point 0.583077049009
- http://pro.imdb.com/rg/maindetails-title/nconst-pro-header-link/name/nm0000129/
  similar point 0.888101142431
- http://pro.imdb.com/
  similar point 0.598701293974
- http://www.amazon.com/exec/obidos/redirect-home/internetmoviedat

similar point 0.106569961628

- https://secure.imdb.com/register/subscribe?c=a394d4442664f6f6475627
  similar point 0.598701293974
- http://www.imdb.de
  similar point 0.224874420559
- http://www.imdb.pt
  similar point 0.346390312595
- http://www.imdb.es
  similar point 0.262768289009

6.2 .8 การทดลองด้วยคำค้นคำว่า Tom Hanks

เว็บไซต์ที่เลือกเป็นเว็บไซต์ตั้งต้นคือ http://www.myspace.com/tomhanks

ผลลัพธ์การเรียงลำดับด้วยวิธี Cosine Similarity

- http://profile.myspace.com/Modules/Applications/Pages/Canvas.aspx?appId=135020&appParams=%7B%22pagename%22%3A%22history%3Ffriend_id%3D190658759%22%2C%22_rye%22%3A%22ms-gifts-profilev1-viewMy-Gifts-click%22%7D
  similar point 1.0
- http://www.myspace.com/pressroom
  similar point 1.0
- http://searchservice.myspace.com/index.cfm?fuseaction=sitesearch.friendfinder
  similar point 1.0
- http://viewmorepics.myspace.com/index.cfm?fuseaction=user.viewAlbums&friendID=190658759
  similar point 0.682355225185
- http://www.myspace.com/index.cfm?fuseaction=InternationalMap
  similar point 0.833487604167

ผลลัพธ์การเรียงลำดับด้วยวิธี K-NN

- http://www.myspace.com/index.cfm?fuseaction=InternationalMap
  similar point 0.312747498871
- http://www.myspace.com/pressroom

similar point 0.065573853562

- http://profile.myspace.com/Modules/Applications/Pages/Canvas.aspx?appId=135020&appParams=%7B%22pagename%22%3A%22history%3Ffriend_id%3D190658759%22%2C%22_rye%22%3A%22ms-gifts-profilev1-viewMy-Gifts-click%22%7D
  similar point 0.038272496748
- http://searchservice.myspace.com/index.cfm?fuseaction=sitesearch.friendfinder
  similar point 0.06488029408
- http://blogs.myspace.com/index.cfm?fuseaction=blog.view&friendId=190658759&blogId=282083675
  similar point 0.517864471428
- http://blogs.myspace.com/index.cfm?fuseaction=blog.view&friendId=190658759&blogId=282085243
  similar point 0.391630680343

6.2 .9 การทดลองด้วยคำค้นคำว่า Cameron Diaz

เว็บไซต์ที่เลือกเป็นเว็บไซต์ตั้งต้นคือ *http://www.imdb.com/name/nm*0000139/

ผลลัพธ์การเรียงลำดับด้วยวิธี Cosine Similarity

- http://resume.imdb.com/
  similar point 0.142659978145
- http://blogs.myspace.com/index.cfm?fuseaction=blog.view&friendId=190658759&blogId=282083675
  similar point 0.563574559566
- http://www.imdb.de
  similar point 0.362881873159
- http://www.imdb.es
  similar point 0.762090864962
- http://www.imdb.pt
  similar point 0.711906532114

ผลลัพธ์การเรียงลำดับด้วยวิธี K-NN

- http://resume.imdb.com/

similar point 0.602297938664

- http://www.imdb.de

  similar point 0.393697541877

- http://www.imdb.es

  similar point 0.208657885253

- http://www.amazon.com/exec/obidos/redirect-home/internetmoviedat

  similar point 0.231255346861

- http://pro.imdb.com/rg/maindetails-title/nconst-pro-header-link/name/nm0000139/

  similar point 0.9939874841

6.2 .10 การทดลองด้วยคำค้นคำว่า Orlando Bloom

เว็บไซต์ที่เลือกเป็นเว็บไซต์ตั้งต้นคือ *http://www.imdb.com/name/nm*0089217/

ผลลัพธ์การเรียงลำดับด้วยวิธี Cosine Similarity

- http://pro.imdb.com/rg/maindetails-title/nconst-pro-header-link/name/nm0089217/

  similar point 0.678733119561

- https://secure.imdb.com/register/subscribe?c=a394d4442664f6f6475627

  similar point 0.47840024662

- http://pro.imdb.com/

  similar point 0.47840024662

- http://www.amazon.com/exec/obidos/redirect-home/internetmoviedat

  similar point 0.496020454243

- http://resume.imdb.com/

  similar point 0.218501213024

- http://resume.imdb.com

  similar point 0.218501213024

- http://resume.imdb.com/?name=Orlando%20Bloom

  similar point 0.218501213024

- http://www.imdb.fr

  similar point 0.162781125376

ผลลัพธ์การเรียงลำดับด้วยวิธี K-NN

- http://www.imdb.de

similar point 0.282964768677

- http://pro.imdb.com/r/imdb-nav-nb/

  similar point 0.350316344206

- https://secure.imdb.com/register/subscribe?c=a394d4442664f6f6475627

  similar point 0.350316344206

- http://resume.imdb.com

  similar point 0.616502199132

- http://www.imdb.fr

  similar point 0.726030129751

- http://pro.imdb.com/rg/maindetails-title/nconst-pro-header-link/name/nm0089217/

  similar point 1.18498428933

6.3 สรุปผล

จากการทดลองพบว่าผลที่ได้นั้นต่างกันไม่มาก เพราะค่า Cosine Similarity ที่นำมาใช้นั้นมีการปรับค่าเป็น TFIDF เรียบร้อยแล้วทำให้ค่าที่ได้ไม่แกว่งมาก เมื่อเปรียบเทียบกับค่า KNN ทำให้ลำดับที่ออกมานั้นไม่แตกต่างกันมากนัก

ปัญหาของการทดลองคือในบางครั้งนั้น Crawler นั้นไม่สามารถเข้าไปเก็บเว็บไซต์บางเว็บไซต์ได้อาจเป็นเพราะติดต่อไม่ได้หรือ หมดเวลาในการติดต่อก่อน ทำให้เว็บไซต์ที่ได้นั้นมีรายชื่อที่ไม่ตรงกัน

บทที่ 7
บทสรุป

7.1 สรุปผลการพัฒนาระบบงาน

โครงการพัฒนาระบบการค้นหาเว็บไซต์แบบเจาะจงนี้ เป็นการพัฒนาขึ้นเพื่อทำให้ผู้ใช้งานสามารถค้นหาเว็บไซต์ที่ต้องการได้ตรงกับความต้องการมากขึ้น และได้ผลลัพธ์ที่มีการเปลี่ยนแปลงล่าสุดเสมอ ซึ่งได้ผลจากการพัฒนาระบบดังนี้

- สามารถแก้ไขปัญหาที่เกิดจากการเปลี่ยนแปลงข้อมูลของเว็บไซต์ที่เกิดขึ้นบ่อย จนทำให้เสิร์ชเอนจินไม่สามารถทำสารบัญเว็บไซต์ทันกับการเปลี่ยนแปลงของเว็บไซต์ได้ โดยการค้นหาเว็บไซต์ภายในเว็บไซต์ที่สนใจและนำเว็บไซต์นั้นมีเป็นผลลัพธ์ในการค้นหา
- ได้ผลลัพธ์ที่มีความเหมือนกับผลลัพธ์ที่ผู้ใช้งานต้องการเพิ่มขึ้น และผลลัพธ์ที่ได้นั้นเป็นลิงก์ที่ยังสามารถใช้งานได้อยู่
- ได้รับความรู้จากการศึกษาการทำงานของเสิร์ชเอนจิน และวิธีการในการค้นหาข้อมูลที่อยู่ในเอกสาร

7.2 ข้อเสนอแนะ

ระบบที่พัฒนานี้เพื่อทดลองหาความสัมพันธ์ที่อยู่ในเว็บไซต์ที่เชื่อมโยงกันโดยนำคุณสมบัติส่วนนี้มาพัฒนาเครื่องมือที่ใช้ในการค้นหาเว็บไซต์ และเพื่อใช้ในการเรียนรู้วิธีการค้นหาข้อมูลที่อยู่ในเอกสาร ในอนาคตหากต้องการนำไปพัฒนาต่อเพื่อให้ใช้งานจริงจำเป็นต้องปรับปรุงและพัฒนาส่วนต่างๆ ดังนี้

- รองรับภาษาอื่นๆนอกจากภาษาอังกฤษ ระบบที่พัฒนานี้สามารถรองรับภาษาได้แค่ภาษาอังกฤษเท่านั้นเนื่องด้วยแต่ละภาษามีลักษณะการใช้งานที่แตกต่างการทำให้การแปลงคำให้อยู่ในรูปที่ทำการประมวลผลได้นั้นมีความแตกต่างกัน เช่นการตัดคำสำหรับภาษาที่มีการเขียนคำติดกัน
- ปัจจุบันเว็บไซต์ที่มีชื่อเสียงเช่น wikipedia.org หรือเว็บไซต์ที่มีผู้เข้าชมจำนวนมากมักจะมีวิธีการที่ทำให้โปรแกรมที่ไม่ใช่เว็บเบราเซอร์เข้าไปเก็บข้อมูลไม่ได้ทำให้ระบบล่มได้ สำหรับวิธีแก้ไขในปัจจุบันคือใช้ไลบรารีที่จำลองการทำงานของโปรแกรมให้เหมือนกับเป็นเว็บเบราเซอร์ตัวนึงซึ่งในอนาคตการแก้ไขนี้อาจใช้งานไม่ได้

บรรณานุกรม

Adrian Holovaty.2008. **The Definitive Guide to Django Web Development Done Right.** New York : Apress.

Bing Liu . 2007. **Web Data Mining Exploring Hyperlinks : Content , and Usage Data .** Springer

Daniel T.Larose. 2005. **Discovering Knowledge in Data :An Introduction To Data Mining.** John Wiley & Sons,

F.Menczer, R. 2000. **Adaptive retrieval agents: internalizing local context and scaling up to the web.** Machine Learning 39 (2-3) 203-242

Filippo Menczer. 2003. **Complementing search engines with online web mining agents.** Decision support systems 35 195-212

Steven Bird. 2009. **Natural Language Processing with Python.** O'Reilly. ผู้แต่ง 3 คน

Zdravko Markov and Daniel T.Larose. 2007. **Data Mining The Web : Uncovering Patterns in Web content , Structure , And Usage.** New Jersey: John Wiley & Sons

ภาคผนวก ก.

ซูโดโค้ดของโมดูลหลัก

1. ซูโดโค้ดของ Data_Preprocessing

```
def Data_Preprecessing(raw_information):
   #clean html
   import nltk
   clean_html = []
   for text in raw_information :
      clean_html.append({'url':text['url'],'text':nltk.clean_html(text['text'])})

   #make token
   import re
   term = []
   for text in clean_html :
      term.append({'url':text['url'],'text':re.split(r'[^A-Za-z]+|[\d]+',text['text'])})

   #convert all characters to lower case
   lower =  []
   i = 0
   for text in term :
      lower.append({'url':text['url'],'text':[]})
      for k  in text['text']:
         lower[i]['text'].append(k.lower())
      i = i + 1

   #Stemming process
   if use_stemming == 1:
      stemming = []
      i = 0
```

2. ซูโดโค้ดของ Cosine Similarity

```
def use_cosine_similarity(main,compare,max_tf):
    main_web = main['text'][:int(max_tf)]
    compared_web = compare['text']
    similar_value = 0
    sum_main = 0
    sum_compare = 0
    for com in compared_web :
        com_term = com.keys()
        value_com = com[com_term[0]]
        for main in main_web:
            dot_value = 0.0000000000000
            main_term = main.keys()
            value_main = main[main_term[0]]
            if com_term == main_term:
                dot_value = float(value_com)*float(value_main)
                sum_main = sum_main + pow(float(value_main),2)
                sum_compare = sum_compare + pow(float(value_com),2)
            similar_value = similar_value + dot_value

    if sum_main == 0 or sum_compare == 0:
        sum_main = 1
        sum_compare = 1
    cosine_similarity_point = similar_value / math.sqrt(sum_main*sum_compare)
    return cosine_similarity_point
```

ประวัติผู้เขียน

ชื่อผู้เขียน	นายนันทวัฒน์ ไชยรัตน์
สถานที่เกิด	จังหวัดสงขลา
ระดับประถมศึกษา	โรงเรียนอนุบาลสงขลา
ระดับมัธยมศึกษาตอนต้น	โรงเรียนมหาวชิราวุธจังหวัดสงขลา
ระดับมัธยมศึกษาตอนปลาย	โรงเรียนมหาวชิราวุธจังหวัดสงขลา
วุฒิการศึกษาระดับปริญญาตรี	สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง