การตรวจจับเว็บสแปมโดยอาศัยการวิเคราะห์บูสเพจ

นายชาคริต ลิขิตขจร

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2554

WEB SPAM DETECTION BASED ON BOOSTED PAGE ANALYSIS

Mr. Chakrit Likitkhajorn

A Thesis Submitted in Partial Fulfillment of the Requirements

for the Degree of Master of Engineering Program in Computer Engineering

Department of Computer Engineering

Faculty of Engineering

Chulalongkorn University

Academic Year 2011

หัวข้อวิทยานิพนธ์	การตรวจจับเว็บสแปมโดยอาศัยการวิเคราะห์บูสเพจ
โดย	นายชาคริต ลิขิตขจร
สาขาวิชา	วิศวกรรมคอมพิวเตอร์
อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก	ผู้ช่วยศาสตราจารย์ ดร. อรรถสิทธิ์ สุรฤกษ์

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญามหาบัณฑิต

…… คณบดีคณะวิศวกรรมศาสตร์
(รองศาสตราจารย์ ดร.บุญสม เลิศหิรัญวงศ์)

คณะกรรมการสอบวิทยานิพนธ์

……ประธานกรรมการ
(รองศาสตราจารย์ ดร.วันชัย ริ้วไพบูลย์)

……อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก
(ผู้ช่วยศาสตราจารย์ ดร.อรรถสิทธิ์ สุรฤกษ์)

……อาจารย์ที่ปรึกษาวิทยานิพนธ์ร่วม
(ผู้ช่วยศาสตราจารย์ ดร. อานนท์ รุ่งสว่าง)

…… กรรมการภายนอกมหาวิทยาลัย
(รองศาสตราจารย์ ดร. เยาวดี เต็มธนาภัทร์)

…… กรรมการภายนอกมหาวิทยาลัย
(รองศาสตราจารย์ ดร. เยาวดี เต็มธนาภัทร์)

ชาคริต ลิขิตขจร : การตรวจจับเว็บสแปมโดยอาศัยการวิเคราะห์บูสเพจ. (Web Spam Detection based on Boosted Page Analysis) อ. ที่ปรึกษาวิทยานิพนธ์หลัก: ผศ. ดร. อรรถสิทธิ์ สุรฤกษ์,อ. ที่ปรึกษาวิทยานิพนธ์ร่วม: ผศ. ดร. อานนท์ รุ่งสว่าง 54 หน้า.

งานวิจัยในการตรวจจับเว็บสแปมโดยทั่วไปจะมีจุดมุ่งหมายหลักในการค้นหาลักษณะเฉพาะของเว็บที่เป็นเว็บสแปม เนื่องจากว่าเว็บสแปมคือเว็บที่ใช้วิธีการพิเศษในการทำให้เว็บเพจของตนเองได้ลำดับสูงกว่าที่ควร ซึ่งในการทำให้เว็บเพจของตนได้ลำดับสูงนั้นจะต้องทำให้ระบบสืบค้นมองเว็บเพจของตนเองว่าได้รับความนิยมสูง เว็บสแปมจะมีการสร้างเว็บเพจที่มีหน้าที่เพิ่มคะแนนความนิยมของตนเอง ซึ่งเว็บเพจเหล่านี้จะเรียกว่า บูสต์เพจ ดังนั้นจึงได้ทำการพัฒนาระบบการตรวจจับเว็บสแปมโดยเริ่มต้นจากวิเคราะห์และตรวจสอบเว็บเพจที่เป็นบูสเพจ แทนที่จะตรวจจับเว็บเพจที่เป็นเว็บสแปมโดยตรง โดยอาศัยลักษณะโครงสร้างความสัมพันธ์ระหว่างเว็บเพจที่เป็นบูสเพจกับเว็บเพจที่เป็นสแปมเป็นตัวชี้วัด แล้วหลังจากนั้นจึงนำเว็บเพจที่เป็นบูสเพจมาเป็นเครื่องมือช่วยหาเว็บเพจที่เป็นเว็บสแปมโดยดูจากโครงสร้างและความสัมพันธ์ระหว่างเว็บสแปมกับบูสต์เพจ และเพจสแปมกับเพจธรรมดา ผลการทดลองพบว่ามีประสิทธิภาพและความแม่นยำในการตรวจจับในระดับที่ดีเมื่อเปรียบเทียบกับงานวิจัยในการตรวจจับเว็บสแปมอื่น ผลลัพธ์จากการตรวจจับเว็บสแปมโดยการวิเคราะห์บูสเพจให้ผลเป็นที่น่าพอใจ

ภาควิชา....วิศวกรรมคอมพิวเตอร์.....ลายมือชื่อนิสิต .....

สาขาวิชา...วิศวกรรมคอมพิวเตอร์ ก(๒)... ลายมือชื่อ อ.ที่ปรึกษาวิทยานิพนธ์หลัก .....

ปีการศึกษา 2554 ลายมือชื่อ อ.ที่ปรึกษาวิทยานิพนธ์ร่วม .....

5270726021 : MAJOR COMPUTER ENGINEERING

KEYWORDS : SEARCH ENGINE / WEB SPAM / LINK FARM / LINK SPAM / SEARCH ENGINE OPTIMIZATION

CHAKRIT LIKITKHAJORN: WEB SPAM DETECTION BASED ON BOOSTED PAGE ANALYSIS. ADVISOR : ASST. PROF. Ph.D. ATHASIT SURARERKS, CO-ADVISOR : ASST. PROF. Ph.D. ARNON RUNGSAWANG, 54 pp.

Generally, research on web spam detection focus on determining characteristic of web spam pages. Since web spam technique is a technique which make target web pages have higher-than-deserve rank in search engine results, these technique must deceived search engine to make target pages look better than it should be. Web spammer try to make their target pages look popular by create many pages to increase popularity score. These pages are call boosted pages. So we introduced new web spam detection algorithm start from analyzing and detecting boosted pages instead of web spam pages. We use links and relationship between boosted pages and web spam pages to determine boosted pages. Then boosted pages will be used for detecting web spam pages by determining difference in relationship between boosted pages and normal pages and relationship between boosted pages and spam pages. Results show that this algorithm produced good accuracy in detecting web spam compared to other web spam detection research. Determining boosted pages can help improve web spam detection to have higher accuracy.

Department : Computer Engineering Student's Signature .....
Field of Study : Computer Engineering A(2) Advisor's Signature .....
Academic Year : 2554 Co-advisor's Signature .....

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จได้ด้วยความเมตตากรุณา และความช่วยเหลือจาก ผู้ช่วยศาสตราจารย์ ดร. อรรถสิทธิ์ สุรฤกษ์ และ ผู้ช่วยศาสตราจารย์ ดร. อานนท์ รุ่งสว่าง ผู้เป็นอาจารย์ที่ปรึกษาวิทยานิพนธ์ ซึ่งได้คอยให้คำชี้แนะ ความช่วยเหลือ ข้อคิด และแนวทางต่างๆ จนทำให้วิทยานิพนธ์เล่มนี้สำเร็จลุล่วงด้วยดี จึงขอขอบพระคุณเป็นอย่างสูงที่มอบโอกาส ความช่วยเหลือ และความเมตตาให้แก่ผู้วิจัยเป็นอย่างดีเสมอมา

ขอกราบขอบพระคุณรองศาสตราจารย์ ดร. วันชัย ริ้วไพบูลย์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ผู้เป็นประธานกรรมการสอบวิทยานิพนธ์ และ รองศาสตราจารย์เยาวดี เต็มธนาภัทร์ กรรมการสอบวิทยานิพนธ์ ผู้ได้ให้คำแนะนำในการพัฒนาวิทยานิพนธ์ให้มีคุณภาพดียิ่งขึ้น

ขอกราบขอบพระคุณคณาจารย์ภาควิชาวิศวกรรมคอมพิวเตอร์ จุฬาลงกรณ์มหาวิทยาลัย ทุกท่าน ที่ได้มอบความรู้ความเข้าใจในเชิงวิชาการ ที่ทำให้ผู้วิจัยมีโลกทัศน์และความรู้ความสามารถมากขึ้น ความรู้ที่ได้รับมาทำให้ผู้วิจัยสามารถสร้างแนวคิดและพัฒนางานวิจัยจนสำเร็จลุล่วง ทั้งในทางตรงและทางอ้อม จึงต้องขอขอบพระคุณมาในที่นี้

สุดท้ายขอกราบขอบพระคุณบิดา มารดา ครอบครัวและญาติพี่น้อง ที่ได้ให้การสนับสนุนผู้วิจัยและให้กำลังใจเสมอมา รวมไปถึงเพื่อนร่วมวิจัยและเพื่อนพี่น้องทุกท่านที่คอยให้คำปรึกษาและให้กำลังใจเป็นแรงผลักดันจนผู้วิจัยสามารถทำวิทยานิพนธ์ฉบับนี้ได้สำเร็จลุล่วง

สารบัญ

หน้า

สารบัญภาพ

หน้า

สารบัญตาราง

หน้า

บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญของปัญหา

ปัจจุบันการใช้งานโปรแกรมสืบค้น (search engine) ในการค้นคืนเว็บเพจให้ตรงกับความต้องการ กำลังเป็นที่นิยมอย่างแพร่หลาย เนื่องจากมีการพัฒนาทั้งในด้านความถูกต้องของผลลัพธ์ และความรวดเร็วมาอย่างต่อเนื่อง ซึ่งในการค้นคืนเว็บเพจนั้น อันดับของเว็บเพจในกลุ่มผลลัพธ์ มีผลต่อพฤติกรรมการเข้าเว็บเพจของผู้ใช้สูงมาก เว็บเพจที่มีอันดับต้นๆ ในผลลัพธ์การสืบค้น มีความเป็นไปได้ที่ผู้ใช้จะเข้าไปอ่านข้อมูลสูงกว่าเว็บเพจที่อยู่ในอันดับรองลงมาเป็นอย่างมาก ทำให้ผู้สร้างเว็บเพจมีความต้องการที่จะให้เว็บของตัวเองอยู่ในอันดับต้นๆ ของผลการค้นคืนจากการค้นหาต่างๆ จึงเป็นที่มาที่มีการกระทำที่พยายามจะทำให้อันดับของเว็บเพจเป้าหมายในผลสืบค้นของเว็บเพจเป้าหมายสูงขึ้นโดยไม่คำนึงถึงความเหมาะสมและคุณภาพของเนื้อหาตามความเป็นจริง เพื่อเพิ่มจำนวนผู้ใช้ที่เข้ามาอ่านเว็บเพจของตนเองให้มากที่สุด ซึ่งเราเรียกการกระทำเหล่านี้ว่า การทำเว็บสแปม (web spam)

ในการทำเว็บสแปมนั้นมีเทคนิคที่นิยมใช้ 2 ประเภท คือเทคนิคคอนเทนท์สแปม (content spam) และเทคนิคลิงก์สแปม (link Spam) โดยเทคนิคคอนเทนท์สแปมนั้นคือการพยายามทำให้เว็บเพจเป้าหมาย สามารถค้นคืนได้จากคำสืบค้นจำนวนมาก โดยไม่สนใจว่าคำสืบค้นเหล่านั้นมีความเกี่ยวข้องกับเนื้อหาของเว็บเพจหรือไม่ เช่น การเพิ่มคำค้นหาจำนวนมากลงในเว็บเพจของต้น ส่วนเทคนิคการทำลิงก์สแปม เป็นเทคนิคการทำการโยกย้ายลิงก์ระหว่างเว็บเพจเพื่อให้เป้าหมายได้รับคะแนนความสำคัญ (ranking score) เพิ่มสูงขึ้น การทำเว็บสแปมนั้นทำให้ประสิทธิภาพในการสืบค้นด้วยโปรแกรมค้นหาลดลง เนื่องจากผลการสืบค้นที่ได้จะประกอบด้วยเว็บเพจที่ไม่เกี่ยวข้องจำนวนมาก ซึ่งการตรวจจับคอนเทนท์สแปมโดยใช้มนุษย์ตรวจสอบสามารถทำได้ง่าย ขณะที่การตรวจสอบลิงก์สแปมโดยใช้มนุษย์ทำได้ยาก และไม่มีประสิทธิภาพ ดังนั้นการค้นหาวิธีการตรวจจับลิงก์ฟาร์มอัตโนมัติจึงเป็นปัญหาที่สำคัญปัญหาหนึ่ง

การตรวจจับลิงก์ฟาร์มอัตโนมัติมีงานวิจัยที่เสนอวิธีการต่างๆ กันมากมาย เช่น การนำข้อมูลคุณสมบัติต่างๆ ของเว็บเพจและลิงก์มาผ่านอัลกอริธึมการเรียนรู้ด้วยเครื่องเช่นซัพพอร์ตเวกเตอร์แมชชีนหรือต้นไม้ตัดสินใจ [1,2] การพิจารณาหาเว็บกราฟขนาดย่อยที่มีความหนาแน่นของลิงก์สูง [3,4,5] อีกวิธีหนึ่งที่เป็นที่นิยมในการค้นหาสแปมคือการเริ่มต้นสืบค้นจากเซ็ตเริ่มต้น (seed set) ของเว็บสแปมจำนวนไม่มาก แล้วนำไปขยายผลค้นหาคะแนนของเว็บสแปมผ่านทฤษฎีความน่าจะเป็น [6,7,8] วิธีนี้จะเริ่มต้นจากการสร้างเซ็ตเริ่มต้นที่มั่นใจว่าเป็น

สแปมแน่นอนจำนวนหนึ่ง แล้วจึงดูว่าจากเซ็ตเริ่มต้นนั้นมีลิงก์ออกไปหาเว็บเพจใดบ้าง เพื่อนำมาคำนวณความเป็นไปได้ในการที่เว็บเพจที่ถูกลิงก์ถึงจะเป็นสแปม

ปัจจุบันวิธีการจัดอันดับเว็บเพจที่นิยมใช้กันจะใช้การอ้างคะแนนจากลิงก์เป็นหลักหรือที่มีชื่อว่าคะแนนเพจแร็งก์ (PageRank) [9] โดยมีแนวคิดว่าเว็บเพจที่มีลิงก์เข้ามาคือเพจที่มีคุณภาพ และเพจที่มีคุณภาพย่อมลิงก์ไปหาเพจที่มีคุณภาพเช่นเดียวกัน ด้วยแนวคิดนี้เมื่อสร้างเว็บกราฟและคำนวนคะแนนทั้งหมดออกมาจะทำให้เราสามารถจัดอันดับคุณภาพของเว็บเพจได้ ดังนั้นเพจที่มีคะแนนสูงในการจัดอันดับเพจด้วยหลักการนี้ คือเพจที่ได้รับลิงก์เข้าจากเพจที่มีคุณภาพสูง หรือได้รับลิงก์เข้าจากเพจที่มีคุณภาพปานกลางหรือต่ำเป็นจำนวนมากเพียงพอ

โดยทั่วไป รูปแบบของการทำลิงก์ฟาร์มจะทำโดยการใช้เว็บเพจขนาดเล็กจำนวนมากที่ผู้ทำสแปม(spammer) สามารถควบคุมได้ชี้เข้ามายังเว็บเพจเป้าหมายที่ต้องการสแปม เย ดู และคณะ (Ye Du) [10] สามารถค้นหารูปแบบทั่วไปของการทำลิงก์สแปมได้ ซึ่งจะเรียกรูปแบบการจัดลิงก์นี้ว่าสแปมฟาร์ม (spam farm) และ กิตติคุณ ชอบธรรม [11] ได้นำรูปแบบของลิงก์สแปมมาจัดเป็นไวยากรณ์กราฟ เพื่อจัดเว็บเพจที่มีลักษณะตรงกับไวยากรณ์กราฟว่าเป็นสแปม ทั้งนี้ ถึงแม้ว่าเราจะทราบรูปแบบของการทำลิงก์สแปมโดยทั่วไปแล้ว ก็ยังเป็นการยากที่จะแยกระหว่างเพจที่ไม่ใช่สแปมที่มีลิงก์เป็นจำนวนมาก และเพจที่เป็นสแปม ถึงแม้ว่าปัจจุบันสามารถค้นหารูปแบบของสแปมฟาร์มเหมาะสมที่สุด (optimal spam farm) ได้ แต่ก็ยังไม่สามารถสรุปได้ว่าเพจที่มีรูปแบบไม่ตรงกับ สแปมฟาร์มเหมาะสมที่สุดจะเป็นเพจที่ไม่ใช่สแปมเสมอไป ซึ่งพบว่าเพจสแปมโดยทั่วไปนั้นไม่ได้อยู่ในรูปของสแปมฟาร์มเหมาะสมที่สุด ดังนั้น ถึงแม้ว่าเราจะพอรู้แล้วว่าการสร้างสแปมทำอย่างไร แต่กระบวนการย้อนกลับ (ตรวจจับ) ยังเป็นปัญหาใหญ่อยู่

เนื่องจากเว็บเพจในอินเทอร์เน็ตนั้นมีการลิงก์ไปมาหาสู่กันเป็นจำนวนมาก ดังนั้นจึงเป็นไปได้สูงที่การเริ่มต้นจากเซ็ตของเว็บจำนวนหนึ่งแล้วเดินตามลิงก์ทั้งหมดที่มี จะสามารถเข้าถึงเว็บเพจได้ทั่วถึงทั้งอินเทอร์เน็ต มีงานวิจัยกลุ่มหนึ่ง [3,4,5] ใช้หลักการนี้ในการตรวจจับเว็บสแปม โดยตั้งแนวคิดว่าเว็บสแปมนั้นจะชี้ลิงก์ไปสู่เว็บสแปมด้วยกันเสียเป็นส่วนมาก และเว็บดีนั้นมักจะชี้เข้าหาเว็บที่ดีด้วยกัน ดังนั้น หากเริ่มต้นค้นหาจากกลุ่มของเว็บจำนวนหนึ่งที่เรามั่นใจว่าเป็นสแปมแล้ว ก็เป็นไปได้สูงที่เราจะสามารถเข้าถึงเว็บสแปมทั้งหมดในอินเทอร์เน็ต งานวิจัย [12] ใช้ค่าความน่าจะเป็นเมื่อจำลองสถานการณ์ว่าผู้ใช้เข้าเว็บที่เป็นสแปมที่เราทราบอยู่แล้ว แล้วไปยังเพจต่อไปโดยเลือกไฮเปอร์ลิงก์อย่างสุ่มเพื่อเดินทางไปทั่วเวิร์ลไวด์เว็บ เพจในที่มีค่าความน่าจะเป็นสูงที่จะถูกเข้าถึงเพจนั้นน่าจะเป็นเพจสแปม งานวิจัย [7] ใช้การคำนวณหาอัตราส่วนคะแนนของแต่ละเพจว่าคะแนนที่เพจนั้นได้รับมาจากต้นลิงก์ที่เป็นเพจสแปมหรือเพจปกติมากกว่ากัน หากได้รับคะแนนจากเพจสแปมเป็นอัตราส่วนมากกว่าค่าเริ่มต้นค่าหนึ่งแล้ว เว็บเพจนั้นเป็นสแปม งานวิจัย [6] ใช้การคำนวนหาค่าความดีของเพจ โดยทำการคำนวนคะแนนของเพจโดย

เริ่มต้นจากกลุ่มของเพจจำนวนหนึ่งที่มีความมั่นใจว่าเป็นเพจดี หากเพจใดได้รับคะแนนสูงก็เป็นไปได้สูงที่จะไม่เป็นเพจสแปม งานวิจัย [8] มองงาน [6] ในมุมกลับแล้วคิดคำนวนค่าความเป็นสแปมของเพจ

เนื่องจากในงานวิจัย [10] นั้นได้แสดงโครงสร้างของลิงก์ฟาร์มและพบว่าคะแนนที่เพิ่มขึ้นของเว็บสแปมนั้นได้รับมาจากสองส่วนหลักๆ (1) ได้รับเพิ่มขึ้นมาจากเว็บเล็กๆ ที่เจ้าของสแปมสร้างขึ้นเพื่อใช้ในการเร่งคะแนนของเว็บสแปมซึ่งได้ให้นิยามไว้ว่าเป็น บูสต์เพจ (Boost Page) (2) ได้รับเพิ่มมาจากการพยายามทำให้ลิงก์เข้าหาเว็บสแปมนั้นปรากฏอยู่บนเว็บที่มีคะแนนสูงหรือเรียกว่าลิงก์ไฮแจกค์ (Hijack Link) ซึ่งในงานวิจัยเว็บสแปมนั้นโดยทั่วไปจะไม่คำนึงถึงปัญหาลิงก์โจรกรรม เพราะงานวิจัยเว็บสแปมนั้นมองว่าการป้องกันไม่ให้ลิงก์ไม่พึงประสงค์ปรากฏอยู่ในเว็บไซต์ที่มีคะแนนสูงนั้นจะเป็นงานวิจัยอีกกลุ่มหนึ่ง และเป็นหน้าที่ของเจ้าของเว็บไซต์เหล่านั้นที่จะต้องป้องกัน ดังนั้น ปัญหาหลักในงานวิจัยเว็บสแปมปัจจุบันคือการป้องกันการเพิ่มคะแนนจากบูสต์เพจ

จากที่กล่าวมาทั้งหมดจะพบว่าโดยปกติแล้ว การค้นหาวิธีตรวจจับเวบสแปมนั้นจะเน้นไปที่การดูลักษณะ (Characteristic) ของตัวเว็บที่เป็นสแปม เช่น เนื้อหา ลิงก์เข้า ลิงก์ออก เพื่อตรวจจับตัวเว็บที่เป็นสแปมโดยตรง แต่เนื่องจากเราทราบแล้วว่าการเพิ่มคะแนนของเว็บที่เป็นสแปมนั้นมีการใช้บูสต์เพจที่มีคะแนนน้อยเป็นจำนวนมาก ดังนั้น แทนที่เราจะไปดูลักษณะของตัวเว็บที่เป็นสแปมโดยตรง เราสามารถตรวจจับเว็บสแปมได้โดยการดูลักษณะเฉพาะของบูสต์เพจที่ใช้ในการเร่งคะแนนเว็บสแปมได้

เนื่องจากว่าวิธีการคิดคะแนนความนิยมของเว็บเพจในปัจจุบันนั้นจะต้องใช้เว็บเพจเล็กที่สร้างขึ้นมาใหม่เป็นจำนวนมากในการชี้เข้าหาเพจที่ต้องการทำสแปม ดังนั้นการทำเว็บสแปมย่อมต้องอาศัยบูสต์เพจเป็นจำนวนมาก เป็นการยากที่ผู้ทำสแปมจะสร้างความหลากหลายให้แก่บูสต์เพจที่ต้องสร้างขึ้นเป็นจำนวนมากในระยะเวลาอันสั้น นอกจากนั้นการจะซ่อนลักษณะบูสต์เพจที่มีจำนวนมากให้หนีพ้นจากระบบตรวจจับต่างๆ ย่อมทำได้ยากกว่าการซ่อนลักษณะของตัวเว็บสแปมที่มีเพียงเว็บเพจเดียว จากเหตุผลที่กล่าวมาผู้วิจัยจึงเห็นว่าการตรวจจับเว็บสแปมโดยตรวจจับที่บูสต์เพจจะให้ผลลัพธ์ที่ดีกว่าการตรวจจับจากเพจที่เป็นเว็บสแปมโดยตรง

งานวิจัยนี้ได้นำเสนอเทคนิคการตรวจจับเว็บเพจที่ทำหน้าที่เป็นบูสต์เพจโดยอาศัยการตรวจสอบลักษณะความสัมพันธ์และโครงสร้างลิงก์ของเว็บเพจ และเมื่อได้บูสต์เพจแล้วจึงนำไปใช้ในการตรวจจับเว็บสแปม โดยอาศัยโครงสร้างลิงก์และความสัมพันธ์ระหว่างตัวบูสต์เพจกับเพจต่างๆ โดยนำไปหักลบกับความน่าเชื่อถือที่เกิดจากการวิเคราะห์ความสัมพันธ์ระหว่างเพจปกติที่น่าเชื่อถือ กับเพจที่กำลังตรวจสอบว่าเป็นสแปมหรือไม่ เพื่อให้ได้ผลลัพธ์ที่ดียิ่งขึ้นในแง่ของความคงทนต่อการปรับระดับความละเอียดของระบบตรวจจับ และความแม่นยำในการตรวจจับเว็บ

สแปม ซึ่งแนวทางดังกล่าวนี้ยังมีประโยชน์ที่สามารถนำไปประยุกต์ใช้ได้กับระบบตรวจจับเว็บสแปมเพื่อพัฒนาความแม่นยำในการตรวจจับได้

1.2 วัตถุประสงค์งานวิจัย

1. **พัฒนาระบบตรวจจับเว็บเพจที่ทำหน้าที่เป็นตัวเร่งคะแนนให้แก่เว็บสแปม**
2. พัฒนาระบบตรวจจับเว็บสแปมที่อาศัยการต่อยอดจากข้อมูลบูสต์เพจ
3. เปรียบเทียบผลลัพธ์ระหว่างระบบการตรวจจับเป็นสแปมที่นำเสนอ กับงานวิจัยอื่น

1.3 ขอบเขตของการวิจัย

1. เสนออัลกอริธึมที่ใช้ในการตรวจจับเว็บสแปม โดยอาศัยการตรวจสอบโครงสร้างระหว่างบูสต์เพจและเว็บสแปมเพจในระดับโฮส
2. ทำการทดสอบประสิทธิภาพกับเว็บกราฟในชุดข้อมูลมาตรฐานที่เก็บโดยฝ่ายวิจัยยาฮู จากโดเมน .uk ในปี 2006
3. เปรียบเทียบประสิทธิภาพการทำงานกับอัลกอริธึมของงานวิจัยที่เกี่ยวข้อง

1.4 ประโยชน์ที่ได้รับ

1. สามารถทราบถึงลักษณะเฉพาะตัวเชิงโครงสร้างของบูสต์เพจ
2. สามารถทราบถึงลักษณะความสัมพันธ์เชิงโครงสร้างของบูสต์เพจและเว็บสแปม
3. สามารถทราบถึงความสัมพันธ์เชิงกว้างระหว่างเว็บสแปมแต่ละเว็บได้
4. สามารถตรวจจับเว็บสแปมที่ทำให้คุณภาพของการสืบค้นเว็บเพจลดลงได้

1.5 วิธีดำเนินการวิจัย

1. ศึกษาลักษณะโครงสร้างของเว็บสแปม และวิธีการทำสแปมที่ใช้ในปัจจุบัน
2. ศึกษางานวิจัยที่นำเสนอวิธีการตรวจจับเว็บสแปมต่าง
3. พัฒนาระบบการตรวจจับบูสต์เพจ
4. พัฒนาระบบการตรวจจับเว็บสแปมโดยอาศัยบูสต์เพจ
5. ทดสอบประสิทธิภาพของระบบที่พัฒนาทั้งสองระบบโดยดูจากความแม่นยำในการตรวจสอบ
6. แก้ไขและปรับปรุงระบบการตรวจจับเว็บสแปม

7. วิเคราะห์ผลการทดลอง สรุปผลการวิจัยและตีพิมพ์งานวิจัย
8. เรียบเรียงและจัดทำวิทยานิพนธ์

1.6 ผลงานตีพิมพ์จากงานวิจัย

ส่วนหนึ่งของงานวิทยานิพนธ์นี้ ได้รับการตีพิมพ์เป็นบทความทางวิชาการในหัวข้อ “A Novel Approach for Spam Detection Using Boosting Pages” โดย ชาคริต ลิขิตขจร , อรรถสิทธิ์ สุรฤกษ์ และ อานนท์ รุ่งสว่าง ในการประชุมวิชาการ Eighth International Joint Conference on Computer Science and Software Engineering (JCSSE) 2011 ณ มหาวิทยาลัยมหิดล วิทยาเขตศาลายา นครปฐม เมื่อวันที่ 11-13 พฤษภาคม พ.ศ. 2554

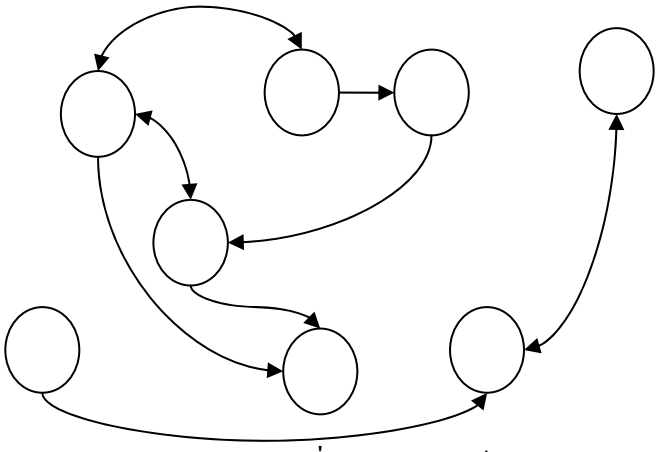
บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 ทฤษฎีที่เกี่ยวข้อง

2.1.1 เว็บกราฟ

เว็บกราฟ (web graph) หมายถึงกราฟระบุทิศทาง G = (V, E) โดยที่ V นั้นเป็นเซ็ตของโหนดที่แทนด้วยหน้าเว็บเพจหรือเว็บโฮส และ E คือเส้นเชื่อมจากเพจหรือเว็บโฮสจากโหนดหนึ่งไปสู่อีกโหนดหนึ่ง ซึ่งหมายถึงลิงก์อย่างน้อยหนึ่งลิงก์ จากเว็บเพจหรือโฮส V_1 ไปสู่ เว็บเพจหรือโฮส V_2 โดยแต่ละเส้นเชื่อมไม่มีน้ำหนักและไม่มีวงวนชี้เข้าโหนดตัวเองในแต่ละโหนดนั้นอาจมีทั้งลิงก์ที่ชี้เข้ามา (inlink) และลิงก์ที่ชี้ออกจากโหนด (outlink) โดยจำนวนลิงก์ที่ชี้เข้าสู่โหนดจะเรียกว่า อินดีกรี(indegree) และจำนวนลิงก์ที่ชี้ออกจากโหนดจะเรียกว่า เอาท์ดีกรี (outdegree) เว็บกราฟนี้เป็นแบบจำลองที่ใช้กันกว้างขวางเพื่อช่วยในการศึกษาความสัมพันธ์ระหว่างเว็บเพจที่มีความซับซ้อนและมีขนาดใหญ่ เช่น การทำเพจแรงค์ [9] หรือการศึกษาเทคนิคการทำลิงก์ฟาร์ม [10]



ภาพที่ 2.1 ตัวอย่างเว็บกราฟ

2.1.2 ลิงก์ฟาร์ม

คือกลุ่มของเว็บเพจที่อยู่ภายใต้การควบคุมของผู้สร้างสแปม โดยมีจุดประสงค์ในการเพิ่มคะแนนให้กับเว็บเพจเป้าหมาย ในลิงก์ฟาร์มนี้ จะมีลิงก์ไปมาหาสู่กันอย่างหนาแน่น และมีจำนวนโหนดที่จำกัด ทำให้เมื่อมีผู้ใช้ที่มีพฤติกรรมการท่องเว็บตามสมมติฐานของการคิดคะแนนเพจแรงก์ [9] ท่องเว็บเข้ามาสู่โหนดที่อยู่ในลิงก์ฟาร์ม ย่อมมีความน่าจะเป็นในการท่องเว็บออกจากเว็บที่บรรจุในลิงก์ฟาร์มน้อยมาก จึงส่งผลให้เมื่อคำนวนคะแนนเพจแรงก์แล้วจะได้ค่าคะแนนสูงกว่าปกติ ทำให้เว็บเพจเป้าหมายอยู่ในอันดับต้นๆ ของผลการสืบค้น การสร้างลิงก์ฟาร์มโดยปกติจะสร้างโดยโปรแกรมอัตโนมัติมักไม่มีเนื้อหาที่เป็นประโยชน์ต่อผู้อ่าน และจากการศึกษางานวิจัย

[10,13] พบว่าเว็บเพจปกติจะไม่ชี้ลิงก์เข้าไปหาลิงก์ฟาร์ม แต่ลิงก์ฟาร์มมีสิทธิ์ที่จะชี้ไปยังเพจใดๆ ก็ได้ นอกจากนี้วิธีการเพิ่มคะแนนลิงก์ฟาร์มยังสามารถนำลิงก์อื่นๆ เข้ามาช่วยได้ ดังนั้น ในมุมมองของผู้สร้างสแปมจะมองเว็บเพจออกเป็น 3 กลุ่ม

1. กลุ่มเว็บที่ไม่สามารถเข้าถึงได้ (inaccessible) คือกลุ่มของเว็บเพจที่ผู้สร้างสแปมไม่สามารถควบคุมให้ชี้มายังลิงก์ฟาร์มของตนได้
2. กลุ่มเว็บที่สามารถเข้าถึงได้ (accessible) คือกลุ่มของเว็บเพจที่ผู้สร้างสแปมสามารถทำให้ชี้ลิงก์มายังลิงก์ฟาร์มของตนได้ มักจะเป็นเว็บไซต์จำพวกเว็บบอร์ด บล็อก หรือเว็บที่มีลักษณะเป็นสาธารณะสามารถโพสลิงก์ได้
3. กลุ่มเว็บที่เป็นของผู้สร้างสแปม หรือกลุ่มที่เรียกว่า ลิงก์ฟาร์ม ซึ่งประกอบด้วยโหนดจำนวนหนึ่ง ซึ่งประกอบด้วยเพจเป้าหมาย (target page) และบูสต์เพจ (boost page) ที่สร้างมาเพื่อเร่งคะแนนให้แก่เพจเป้าหมาย แต่ละโหนดในสแปมฟาร์มประกอบกันด้วยลิงก์ที่หนาแน่น

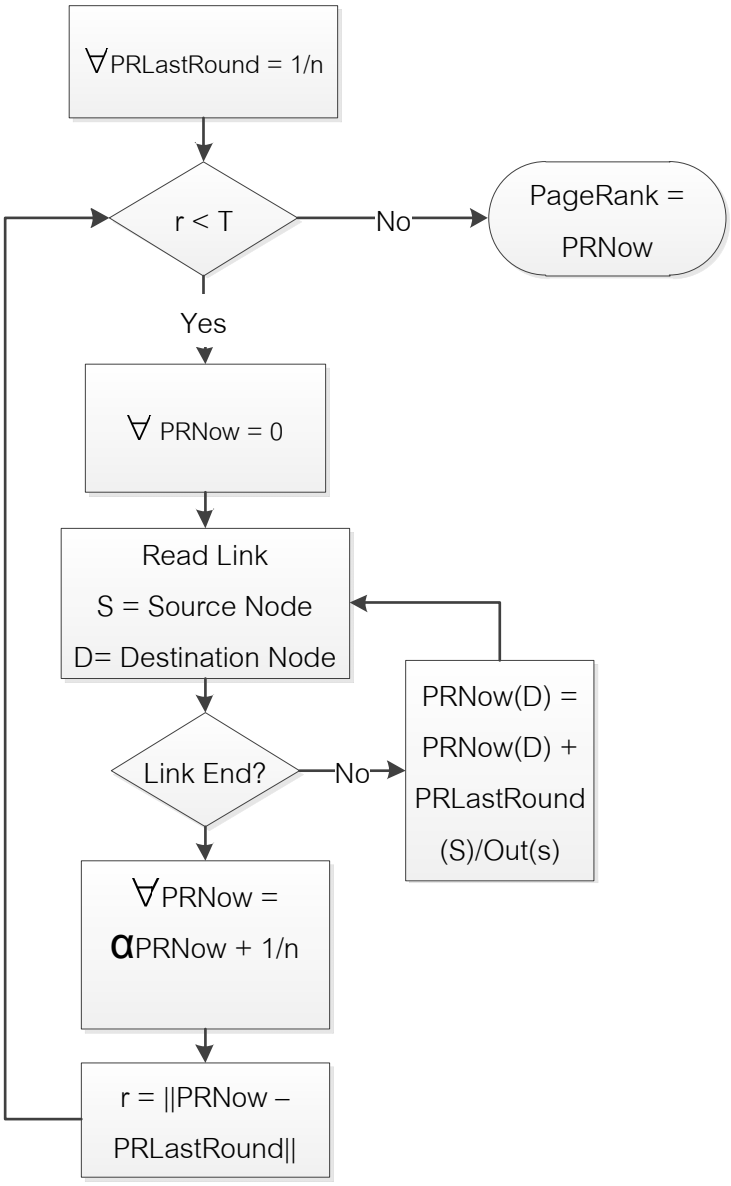
2.1.3 เพจแรงก์

เพจแรงก์ (PageRank) [10] คืออัลกอริธึมที่ใช้ในการคิดค่าความสำคัญของเว็บเพจใดๆ เพื่อใช้ในการจัดเรียงอันดับผลการสืบค้นเว็บจากคำค้นหา ซึ่งในปัจจุบันเป็นที่นิยมใช้กันในโปรแกรมค้นหา (search engine) ทั่วไปเช่น กูเกิ้ล (Google) วิธีการคิดคะแนนเพจแรงก์มีแนวคิดมาจากแบบจำลองการเดินสุ่มตามทฤษฎีมาร์คอฟเชน ซึ่งเปรียบเสมือนกับการคิดค่าความน่าจะเป็นที่ผู้ใช้จะเข้าถึงเว็บเพจต่างๆ ในเว็บกราฟโดยเริ่มต้นจากเว็บเพจต้นทางอย่างสุ่ม แล้วเดินทางไปตามลิงก์ออกจากเว็บนั้นด้วยความน่าจะเป็น α (แอลฟา) ค่าหนึ่ง และเดินทางกระโดดออกจากเว็บไปหาเว็บอื่นที่ไม่เกี่ยวข้องกันเลยอย่างสุ่มด้วยความน่าจะเป็น 1-α ส่วนในแต่ละลิงก์ออกนั้นมีความน่าจะเป็นในการกระโดดไปในแต่ละลิงก์เท่าๆ กัน เมื่อคิดตามนี้ก็จะสามารถหาความน่าจะเป็นที่ผู้ใช้จะเดินทางเข้าไปสู่แต่ละเว็บได้ โดยเว็บเพจที่มีค่าคะแนนเพจแรงก์สูงนั้นจะแสดงผลเป็นลำดับต้นๆ ในผลสืบค้น ซึ่งจุดแข็งของเพจแรงก์ คือ การเพิ่มคะแนนเพจแรงก์อย่างมีนัยสำคัญจำเป็นจะต้องมีลิงก์เข้าจากเพจที่มีคะแนนเพจแรงก์ต่ำเป็นจำนวนมาก หรือมีลิงก์เข้าจากเพจที่มีคะแนนเพจแรงก์สูง

เมื่อกำหนดค่าความน่าจะเป็นในการกระโดดสุ่ม (random jump probability) α ให้มีค่าภายใน [0,1] (โดยปกติจะอยู่ที่ 0.85) ให้ P_a เป็นคะแนนเพจแรงก์ k ของเพจ a ใดๆ เพจแรงก์ของเพจ x เกิดจากผลรวมของคะแนน เพจแรงก์ ของเพจ y ทั้งหมดที่ชี้เข้าหาเพจ x ดังสมการ 2.1

$$p_x = \alpha \sum_{(y,x)} \frac{p_y}{out(y)} + \frac{1-\alpha}{n} \quad (2.1)$$

เมื่อ out(y) เป็นค่าเอาท์ดีกรีของเพจ y และ n คือจำนวนเพจทั้งหมดในเว็บกราฟ จากสูตรดังกล่าวการคำนวณเพจแรงก์สามารถเขียนเป็นผังงาน (flowchart) ได้ตามภาพที่ 2.2



ภาพที่ 2.2 ผังงานวิธีการคำนวณคะแนนเพจแรงก์

วิธีการคิดคะแนนเพจแรงก์นั้นจะให้คะแนนแก่เพจที่มีลิงก์เข้าเป็นจำนวนมาก หรือมีลิงก์เข้าจากเพจที่มีค่าคะแนนเพจแรงก์สูง โดยคะแนนเพจแรงก์นับเป็นส่วนสำคัญในการจัดลำดับผลค้นคืนจากโปรแกรมค้นหา คะแนนเพจแรงก์นั้นจะถูกใช้เป็นตัวบ่งชี้ความนิยมของเว็บเพจต่างๆ ในผลค้นคืน ซึ่งจะถูกนำไปถ่วงน้ำหนักกับค่าคะแนนความเกี่ยวข้องระหว่างเนื้อหาในเว็บเพจกับคำค้นหาที่ผู้ใช้ป้อน เพื่อที่จะสามารถนำเว็บเพจที่มีคุณภาพสูงสุดที่เกี่ยวข้องกับคำค้นหามากที่สุดและเหมาะสมที่สุด ไว้ในลำดับแรกของผลค้นคืน

2.1.4 การใช้สแปมฟาร์มเพื่อเพิ่มคะแนน PageRank

เย ดู (Ye Du) เหยาหยุน ชิ (Yaoyun Shi) และ ซิน เจ่า (Xin Zhao) [10] ได้นำเสนอวิธีการเร่งคะแนน เพจแรงก์ โดยการใช้สแปมฟาร์ม (spam farm) โดยได้วิเคราะห์ถึงโครงสร้างของสแปมฟาร์มเหมาะสมที่สุด (optimal spam farm) หรือสแปมฟาร์ม ที่สามารถเร่งคะแนน เพจแรงก์ ได้สูงที่สุดเท่าที่เป็นไปได้ เมื่อมองเพจทั้งหมดที่ผู้สร้างสแปมสามารถนำลิงก์เข้าหาเพจที่ต้องการทำสแปมไปบรรจุได้ จะแบ่งเพจเหล่านี้ออกเป็น 3 ประเภทคือ

1. บูสต์เพจ (boost page) เป็นเพจที่ผู้ทำสแปมสร้างขึ้นมาเองเพื่อเร่งคะแนนของเพจเป้าหมาย ซึ่งโดยทั่วไปแล้วจะมีจำนวนมากเพื่อให้สามารถเร่งคะแนนขึ้นได้เป็นจำนวนมาก
2. ไฮแจกค์เพจ (hijack page) เป็นเพจที่ผู้ทำสแปมมิได้สร้างขึ้นเอง แต่สามารถสร้างลิงก์ได้อย่างอิสระ โดยทั่วไปแล้วเพจเหล่านี้จะเป็นเพจที่มีลักษณะเป็นเว็บบอร์ดหรือ โซเชียลเน็ตเวิร์ก (social network) ที่ผู้อ่านสามารถเพิ่มเติมเนื้อหาของเพจได้
3. เพจเป้าหมาย (target page) เป็นเพจเป้าหมายที่ผู้ทำสแปมต้องการเร่งคะแนน

ซึ่งจะได้โครงสร้างที่ใช้ในการเร่งคะแนนเพจแรงก์ออกมา โดยโครงสร้างดังกล่าวมีกฎเกณฑ์พื้นฐานคือ

1. ทุกๆ บูสต์เพจ ในลิงก์ฟาร์ม ต้องชี้ไปยังเพจเป้าหมายเพียงอย่างเดียวเท่านั้น
2. เพจเป้าหมาย ต้องชี้ไปยังบางบูสต์เพจ
3. ทุกๆ เว็บในกลุ่มเว็บที่สามารถเข้าถึงได้ ต้องชี้มายังทุกๆ บูสต์เพจ และเพจเป้าหมาย

ซึ่งหากในบางกรณีกฎที่กล่าวมานี้ไม่สามารถปฏิบัติได้จริง ก็จะใช้โครงสร้างที่คล้ายกัน เช่น หากบูสต์เพจไม่สามารถชี้หาเพจเป้าหมายได้ก็จะให้ชี้เข้าหาบูสต์เพจอื่นๆ ในสแปมฟาร์มแทน

2.2 งานวิจัยที่เกี่ยวข้อง

2.2.1 งานวิจัยค้นหาลิงก์ฟาร์มโดยใช้แบบจำลองการเดินสุ่มจากเซ็ตเริ่มต้น

เบานิ่ง วู (Baoning Wu) และคูมาร์ เชลลาพิลล่า (Kumar Chellapilla) [12] นำเสนอการตรวจจับลิงก์ฟาร์มในปี 2007 โดยการเริ่มต้นจากเซตของเว็บเพจเริ่มต้นที่เป็นลิงก์ฟาร์มโดยใช้มนุษย์ตัดสินใจ และอาศัยแบบจำลองเดินสุ่มในการขยายขอบเขตของลิงก์ฟาร์มโดยที่

$$p(t+1)=\frac{1}{2}\left(I+AD^{t}\right)p(t) \tag{2.2}$$

เมื่อ *A* คือเมทริกซ์ประชิดของเว็บกราฟ *G*

I คือเมทริกซ์เอกลักษณ์

D' คือทรานสิชันเมทริกซ์

ซึ่งความน่าจะเป็นเริ่มต้นของโหนดในลิงก์ฟาร์มเริ่มต้น (*S*) คือ

$$p_0(i)=\begin{cases}1/|S| & : \text{ถ้า } i \in S \\ 0 & : \text{ถ้า } i \notin S\end{cases} \tag{2.3}$$

โดยทำการคำนวนค่าความน่าจะเป็นซ้ำหลายๆ รอบจนค่าความน่าจะเป็นทั้งหมดมีค่าลู่เข้าสู่ค่าๆ หนึ่ง

2.2.2 งานวิจัยค้นหาลิงก์สแปมโดยการประเมินมวลสแปม (spam mass)

โซลตาน จยองยี้ (Zoltan Gyongyi), เฮคเตอร์ การ์เซีย-โมลิน่า (Hector Garcia-Molina), พาเวล เบอร์กิน (Pavel Berkhin) และ แจน ปีเดอร์เซ่น (Jan Pedersen) [7] ได้นำเสนอเทคนิคในการตรวจจับสแปม โดยการใช้ฟังก์ชันเพจแรงก์คอนทริบิวชัน (PageRank contribution) ซึ่งเป็นฟังก์ชันที่สามารถคิดหาได้ว่าในแต่เพจโหนด x ที่ได้รับลิงก์ชี้เข้าจากเพจโหนด y ผ่านทางเดิน (Walk) W นั้น x ได้รับค่าเพจแรงก์จาก ทางเดิน W เป็นจำนวนเท่าไหร่ โดยที่

$$q_y^W = c^k \pi(W)(1-c)v_x, \tag{2.4}$$

เมื่อ

$$\pi(W) = \prod_{i=0}^{k-1} \frac{1}{\mathrm{out}(x_i)}. \tag{2.5}$$

และสามารถคิดคะแนนทั้งหมดที่เพจ x ได้รับจากเพจ y ดังนี้

$$q_y^x = \sum_{W \in W_{xy}} q_y^W .$$

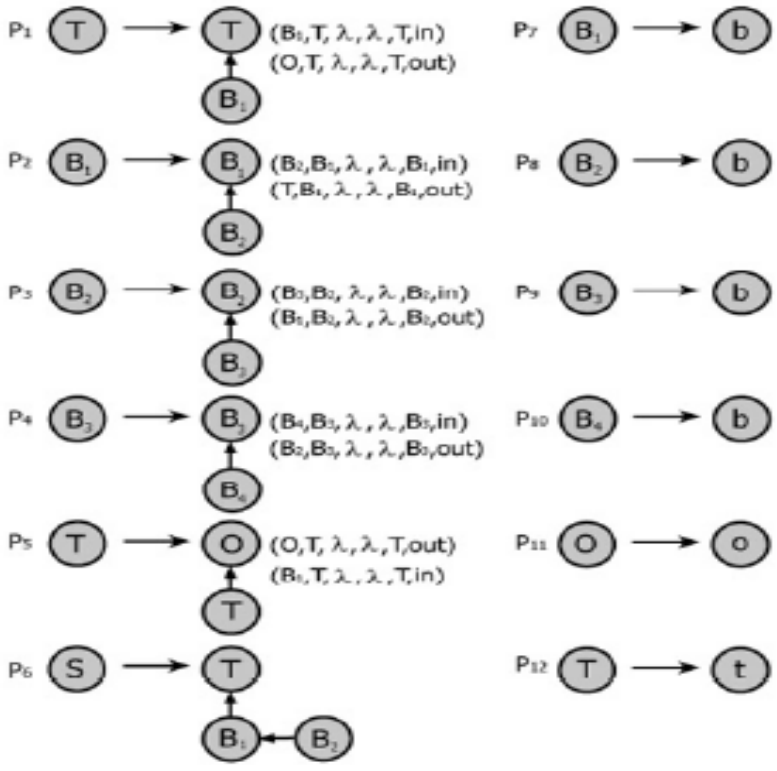
(2.6)

โดยที่ c คือ แดมป์ปิ้งแฟคเตอร์ (damping factor) ที่ใช้ในการคิดคะแนนเพจแรงก์
K คือจำนวนลิงก์เชื่อม ที่มีทั้งหมดในทางเดินนั้นๆ
v คือความน่าจะเป็นที่จะมีการเดินทางผ่านทางเดินนั้นๆ
และ out(x) คือ เอาท์ดีกรีของโหนด x

แล้วจึงทำการเริ่มคิดจากเซ็ตของเพจที่ดีเป็นเซ็ตเริ่มต้น แล้วคิดหาคะแนนของเพจนอกเหนือจากเซ็ตเริ่มต้น ที่ได้รับลิงก์จากเพจที่ดี หากคะแนนที่ได้จากเพจที่ดีมีค่าน้อยกว่าครึ่งหนึ่งของคะแนนเพจแรงก์ ที่แท้จริง ก็จะถือว่าเพจนั้นเป็นสแปม มิฉะนั้นจากถือว่าเพจนั้นเป็นเพจที่ดี เมื่อทำซ้ำวนไปเรื่อยๆ จนค่าที่ได้เริ่มลู่เข้าสู่ค่าใดค่าหนึ่งจึงหยุดการคิดคะแนน

2.2.3 งานวิจัยการตรวจจับลิงก์ฟาร์มโดยใช้ไวยากรณ์กราฟ

เกียรติคุณ ชอบธรรม (Chobtham, K.); อรรถสิทธิ์ สุรฤกษ์ (Surarerks, A.); และอานนท์ รุ่งสว่าง (Rungsawang, A.) [11] ได้นำเสนอการตรวจจับลิงก์ฟาร์ม โดยการกำหนดไวยากรณ์ของกราฟที่สามารถตรวจจับลิงก์ฟาร์ม จากการศึกษารูปแบบของลิงก์ฟาร์มและสแปมฟาร์มเหมาะสมที่สุด ได้ไวยากรณ์ที่ใช้ในการตรวจจับภาพที่ 2.3



ภาพที่ 2.3 แสดงไวยากรณ์กราฟที่ใช้ในการลิงก์ฟาร์ม

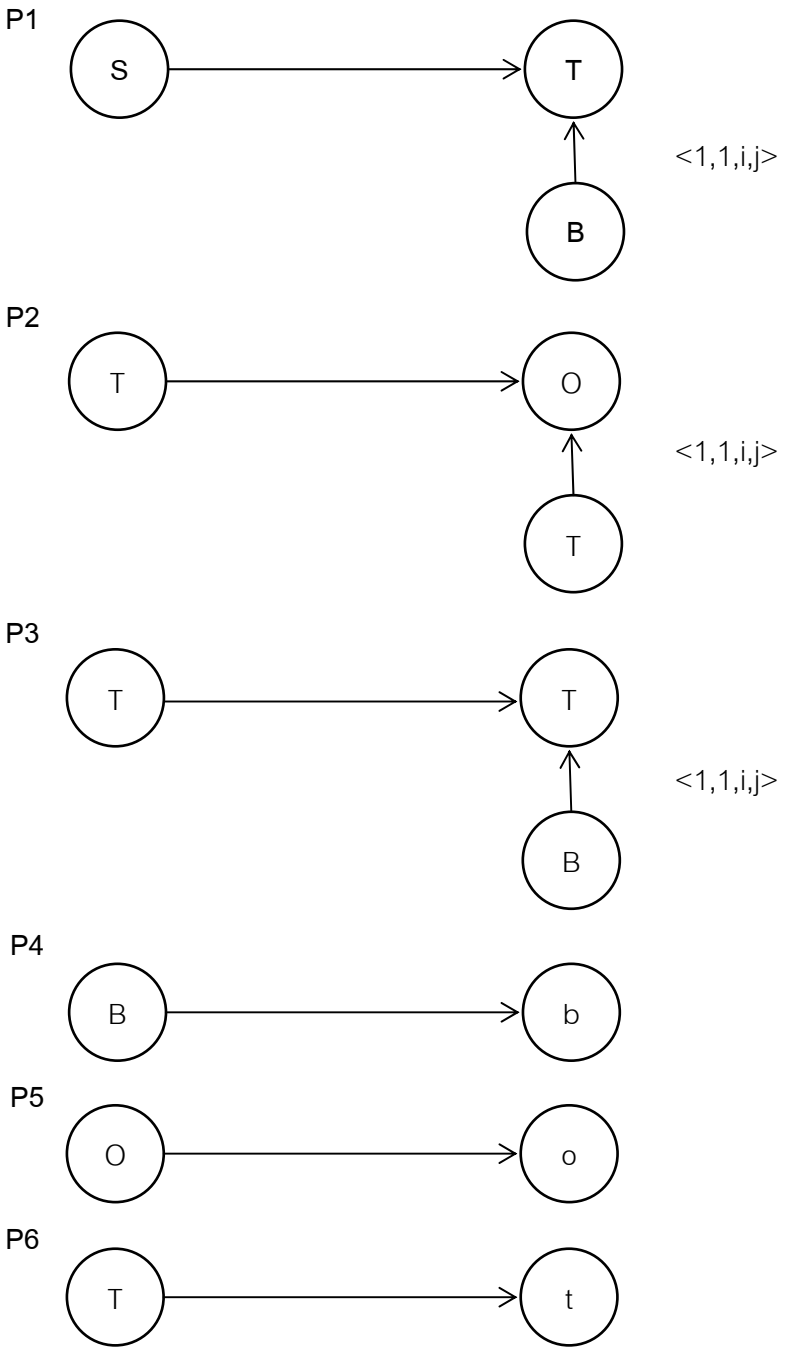
ซึ่งเมื่อได้ไวยากรณ์กราฟดังกล่าวแล้วนำมาเทียบกับข้อมูลเว็บกราฟในระดับโฮสและระดับเพจ ก็จะสามารถตรวจจับเว็บสแปมได้

2.2.4 งานวิจัยการตรวจจับลิงก์ฟาร์มโดยการติดป้ายเส้นเชื่อม

วุฒิชัย วงศ์สารสิน (Wongsarasin W.) อรรถสิทธิ์ สุรฤกษ์ (Surarerks, A.) และ อานนท์ รุ่งสว่าง (Rungsawang, A.) [14] ได้นำเสนอการตรวจจับลิงก์ฟาร์มโดยพัฒนาจากงานวิจัย [11] ด้วยวิธีการเพิ่มป้ายชื่อให้แก่เส้นเชื่อมในเว็บกราฟ แล้วใช้ป้ายชื่อเหล่านั้นเป็นส่วนประกอบในการพิจารณาสร้างไวยากรณ์กราฟเพิ่มเติมที่มีความสามารถในการตรวจจับลิงก์ฟาร์มได้ดีขึ้น โดยคุณสมบัติที่นำมาพิจารณาร่วมนั้นมี 4 ประการ

1. จำนวนลิงก์เข้าและลิงก์ออกของโหนด โดยใช้ค่าอัตราส่วนลอการิธึมในการพิจารณา
2. คะแนน ทรันเคท เพจแรงก์ (Truncated PageRank)
3. คะแนน เพจแรงก์
4. จำนวนการแลกเปลี่ยนลิงก์ (reciprocity) ของโหนด

โดยไวยากรณ์กราฟที่สามารถตรวจจับลิงก์ฟาร์มได้ตามที่ผู้เขียนได้เสนอมีดังนี้



ภาพที่ 2.4 แสดงไวยากรณ์กราฟที่ใช้ตรวจจับสแปมเมื่อมีการติดป้ายชื่อเส้นเชื่อม

กำหนดให้คุณสมบัติเส้นเชื่อม E_i เมื่อ i เป็นเลขจำนวนเต็มและ $1 \leq i \leq 4$ มีค่าดังต่อไปนี้

$$E_t \begin{cases} 1; & \text{เมื่อ } E_{t,t} \geq E_{n,t} \\ 0; & \text{ในกรณีอื่น} \end{cases}$$

โดยที่

$E_{j,1}$ = คะแนนเพจแรงก์ของโหนด j

$E_{j,2}$ = คะแนน ทรันเคท เพจแรงก์ ของโหนด j

$E_{j,3}$ = Log (จำนวนอินดีกรี ของโหนด j) / Log (จำนวนเอาท์ดีกรีของโหนด j)

$E_{j,4}$ = ลิงก์กลับ / จำนวนเอาท์ดีกรีของโหนด j

เมื่อ j ε (t,n) เมื่อ t คือ โหนดเป้าหมาย และ n คือ โหนดเพื่อนบ้าน

เมื่อกำหนดให้คุณสมบัติเส้นเชื่อม $<E_1,E_2,E_3,E_4>$ จะได้ไวยากรณ์กราฟดังภาพที่ 2.3

2.2.5 งานวิจัยตรวจจับเว็บสแปมโดยใช้วิธีการแอนตี้ทรัสต์

[9] วีเจย์ คริชนาน (Krishnan V.) และ ราชมิ ราช (Raj R.) ได้เสนอวิธีการตรวจจับเว็บสแปมโดยใช้ค่าคะแนนแอนตี้ทรัสต์ โดยที่คะแนนแอนตี้ทรัสต์นั้นคิดจากการกระจายคะแนนจากเว็บเพจเริ่มต้นที่ทราบอยู่แล้วว่าเป็นเว็บสแปมจำนวนหนึ่ง เว็บเพจเหล่านี้จะถูกตรวจสอบโดยมนุษย์ หลังจากนั้นจึงนำเว็บเพจเริ่มต้นไปเป็นอินพุตในการคิดคะแนนเพจแรงก์แบบเอนเอียง โดยมีการเพิ่มคะแนนให้มากขึ้นหากพบว่าเพจที่ชี้เข้ามาหาเป็นเพจที่เป็นสแปม ตามสมการที่ 2.7

$$X(t) = d \cdot T^t \cdot X(t-1) + (1-d) \cdot \alpha \qquad (2.7)$$

โดยที่ X(t) คือ คะแนนเพจแรงก์เอนเอียงจากการคำนวณในรอบที่ t

T คือ เมทริกซ์ที่ใช้แทนลิงก์ระหว่างโหนดในกราฟ โดยที่

t_{ij} มีค่าเป็น 1 หากมีลิงก์จากโหนด i ไปยังโหนด j

t_{ij} มีค่าเป็น 0 ในกรณีอื่น

d คือ ค่าคงที่ที่ปรับได้ แทนความน่าจะเป็นในการเที่ยวเว็บเพจโดยมิได้เดินอย่างสุ่ม นิยมใช้ค่า 0.85

α คือ ค่าความน่าจะเป็นที่เอนเอียงให้แก่เพจที่อยู่ในกลุ่มเพจเริ่มต้นที่ทราบว่าเป็นเพจสแปม

จากผลการทดลองพบว่าอัลกอริธึมแอนตี้ทรัสต์จะตรวจจับเว็บสแปมได้อย่างมีประสิทธิภาพเมื่อมีการเลือกเพจที่เป็นสแปมไว้ในเซ็ตเริ่มต้นเป็นจำนวนมาก และมีคะแนนเพจแรงก์สูง

2.2.6 การตรวจจับสแปมแบบทรานส์ดักทิฟ

[2] เดงยอง ซู (Zhou D.), คริสโตเฟอร์ เบิร์จ (Burge C.) และ เตา เตา (Tao.T) ได้นำเสนอการใช้ตัวเรียนรู้แบบซัพพอร์ตเวกเตอร์แมชชีนเข้ามาช่วยในการตรวจจับเว็บสแปม ซึ่งมีขั้นตอนการทำงานดังนี้

กำหนดให้เว็บกราฟ G = (V,E) เป็นกราฟที่เชื่อมกันอย่างหนาแน่น (หากไม่ใช่ให้ทำการตัดโหนดที่ไม่ได้เชื่อมต่ออย่างหนาแน่นออก) โดยมีบางเว็บเพจ S ใน V ถูกกำหนดไว้ก่อนว่าเป็นเว็บเพจประเภทปกติหรือสแปม โหนดที่เหลือจะถูกแยกประเภทโดยวิธีการต่อไปนี้

1. กำหนดฟังก์ชันเดินสุ่มโดยเลือกจากลิงก์เข้า ซึ่งฟังก์ชันเดินสุ่มนี้เป็นตัวกำหนดความน่าจะเป็นในการเข้าสู่โหนดปัจจุบันของผู้เดิน โดยที่

$$p(u,v) = \frac{w(v,u)}{d^{-}(u)}$$

เมื่อ w(v,u) คือน้ำหนักของเส้นเชื่อมจาก v ไปหา u

d^{-} คือผลรวมของน้ำหนักเส้นเชื่อมทั้งหมดที่ชี้เข้าหา u

กำหนดเวกเตอร์ π ซึ่งเป็นไปตามสมการต่อไปนี้

$$\sum_{u \in V} \pi(u)p(u,v) = \pi(v)$$

2. กำหนด P ให้เป็นเมตริกซ์สองมิติที่มีสมาชิกเป็น p(u,v) และกำหนด β เป็นเมตริกซ์ทแยงมุมที่มีสมาชิกเป็น π(u) แล้วกำหนดเมตริกซ์ L ตามสมการต่อไปนี้

$$L = \beta - \alpha\frac{\beta P \mid P^{T}\beta}{2}$$

โดยที่ α เป็นพารามิเตอร์ที่อยู่ในช่วงระหว่าง 0 ถึง 1

3. กำหนดฟังก์ชัน y บนทุกโหนดในกราฟ โดยที่

$$y(v)\begin{cases} 1, \text{หาก } v \text{ เป็น } normal \\ -1, \text{หาก } v \text{ เป็น } spam \\ 0, \text{หาก } v \text{ เป็นโหนด } undecided \end{cases}$$

แล้วจึงนำค่าที่ได้มาแก่ระบบสมการเชิงเส้น

$$L\varphi = \beta y$$

แล้วจำแนกประเภทของโหนดโดยใช้ฟังก์ชัน φ(v)

การทดสอบประสิทธิภาพของอัลกอริธึมนี้ใช้ข้อมูลเว็บกราฟ 2006 ซึ่งมีข้อมูลชุดทดสอบและชุดสอน ผลการทดลองอัลกอริธึมทรานส์ดักทิฟให้ความแม่นยำสูงที่ค่าเรียกคืนต่ำและมีประสิทธิภาพสูงกว่าการตรวจจับแบบแอนตี้ทรัสต์

บทที่ 3

การตรวจจับเว็บสแปมโดยการค้นหาบูสต์เพจ

3.1 ความหมายของเว็บสแปม

โดยทั่วไปแล้ว เว็บสแปมหมายถึงเว็บเพจที่ได้คะแนนอันดับไม่สอดคล้องกับคุณภาพของเนื้อหาในตัวเว็บเพจ หรืออาจมองได้ว่าเว็บสแปมคือเว็บเพจที่มีเนื้อหาไม่เหมาะสมกับลำดับที่ได้ในผลค้นคืน เช่น เว็บเพจที่มีเนื้อหาไปในทางลามกอนาจาร เว็บเพจที่เน้นผลประโยชน์ทางการค้ามากกว่าการให้ข้อมูลข่าวสารความรู้ แต่เนื่องจากความสอดคล้องระหว่างคุณภาพเว็บเพจกับคะแนนอันดับ นั้นยากที่จะนิยามลงไปชัดเจนได้ ผู้ใช้เองก็มีความชื่นชอบที่แตกต่างกันในแต่ละคน ทำให้ผู้ใช้แต่ละคนย่อมกำหนดคุณภาพให้แก่เว็บเพจเดียวกันไม่เท่ากัน ซึ่งเว็บสแปมในมุมของผู้ใช้อาจจะเป็นเว็บเพจที่อันตรายต่อเครื่องคอมพิวเตอร์ เนื่องจากมีชุดโปรแกรมไม่พึงประสงค์ (malicious ware) ติดตั้งอยู่ โดยไม่สนใจในตัวเนื้อหาของเว็บเพจ เนื่องจากเว็บเพจเหล่านี้ผู้ใช้รู้สึกว่ามีคุณภาพต่ำ ดังนั้นคำนิยามของคำว่าเว็บสแปมจึงยังมีความคลุมเครืออยู่ เป็นการยากที่จะนิยามลงไปให้ชัดเจนได้

ในการชี้เฉพาะเจาะจงว่าเว็บใดเป็นเว็บสแปมนั้นเป็นปัญหาสำคัญ ปัญหาหนึ่งคือเว็บสแปมในสายตาของโปรแกรมกับเว็บสแปมในสายตาของผู้ใช้อาจจะไม่ตรงกัน เพราะผู้ใช้นั้นไม่สามารถมองเห็นลักษณะโครงสร้างความสัมพันธ์ระหว่างเว็บเพจได้ชัดเจน เมื่อผู้ใช้ตัดสินเว็บเพจว่าเป็นสแปมหรือไม่ มักจะตัดสินจากเนื้อหาของตัวเว็บเพจ ปัจจัยที่ผู้ใช้พิจารณาว่าเว็บเพจแต่ละเว็บเพจเป็นเว็บสแปมหรือไม่ ก็จะดูที่ว่าคุณภาพของเนื้อหามีความเหมาะสมกับลำดับในผลค้นคืนหรือไม่

การจะกำหนดชี้ชัดลงไปว่าเว็บเพจใดนั้นมีความเหมาะสมหรือไม่เหมาะสมที่จะได้รับลำดับต้นในผลค้นคืนนั้นทำได้ยากและย่อมแตกต่างกันออกไปตามวิจารณญาณของบุคคล เราอาจบอกไม่ได้ชัดเจนว่าเว็บเพจใดนั้นเหมาะสมมากแค่ไหนในเชิงคุณภาพของเนื้อหาในเว็บเพจ แต่หากเรามองแค่เพียงในแง่ที่ว่าเว็บเพจนั้นใช้วิธีการใด ได้รับคะแนนสูงมากพอที่จะอยู่ในลำดับสูงได้เพราะเหตุใด แล้วเหตุผลที่ได้รับคะแนนสูงนั้นตรงกับสภาพความเป็นจริงหรือไม่ หรือมีการพยายามใช้เทคนิคพิเศษเพื่อหลอกลวงโปรแกรมค้นหาให้ได้รับคะแนนสูง ถึงแม้ว่าเราอาจจะบอกไม่ได้ว่าแท้จริงแล้วเว็บเพจนั้นควรได้ลำดับเท่าไหร่ แต่เราก็สามารถชี้ลงไปได้อยากชัดเจนว่าเว็บเพจนั้นใช้วิธีที่ไม่เหมาะสมและควรจะจำแนกเว็บเพจเหล่านี้ออกมาเพื่อเป็นการยุติธรรมแก่เว็บเพจอื่นที่ไม่ได้ใช้เทคนิคพิเศษในการเพิ่มคะแนน

การให้คะแนนของโปรแกรมค้นหานั้นจะแบ่งคะแนนออกเป็นสามส่วน [15] คือ

1. คะแนนความเกี่ยวข้องกับคำค้นหา คะแนนส่วนนี้พิจารณาจากปัจจัยที่ว่าคำที่ผู้ใช้ป้อนเพื่อค้นหา มีความเกี่ยวข้องกับเนื้อหาของเว็บเพจมากน้อยเพียงใด
2. คะแนนความนิยมของเพจ คะแนนส่วนนี้พิจารณาจากปัจจัยที่ว่าเว็บเพจนั้นมีความนิยมมากแค่ไหน ซึ่งอัลกอริธึมที่ใช้คิดคะแนนส่วนนี้ปัจจุบันจะนิยมใช้วิธีการคิดแบบเพจแรงก์เป็นหลัก
3. คะแนน URL ของเว็บเพจ คะแนนส่วนนี้พิจารณาจากที่อยู่ของเว็บเพจนั้น เช่นคุณภาพของโดเมนที่เว็บเพจนั้นอาศัย (โดยบางครั้งอาจจะเทียบกับโดเมนของผู้ค้นหา เช่น หากผู้ค้นหามาจากประเทศไทย ก็จะให้ค่าน้ำหนักแก่โดเมน .th มากกว่าผู้ค้นหาที่มาจากประเทศอื่น) หรือตำแหน่งในเว็บเซิฟเวอร์ เพจที่อยู่ในหน้าหลัก (main page) ก็จะได้คะแนนมากกว่าเพจที่อยู่ในหน้าย่อยที่ซอยลงไปหลายชั้น

เมื่อพิจารณาจากวิธีการอัลกอริธึมการให้คะแนนแล้ว ก็จะมีวิธีการเทคนิคหลอกลวงมากมายที่กล่าวได้ว่าเป็นเทคนิคที่ไม่เหมาะสมอย่างชัดเจนในการทำให้เว็บเพจนั้นขึ้นมาอยู่ในลำดับที่สูง เช่น

1. คะแนนความเกี่ยวข้องกับคำค้นหา สามารถเพิ่มคะแนนได้โดยการเพิ่มคำที่เป็นที่นิยมใช้ในการค้นหาลงไปเป็นจำนวนมากแก่เว็บเพจของตน โดยที่ไม่สนใจว่าคำเหล่านั้นกับเนื้อหาโดยรวมของเว็บเพจมีความเกี่ยวข้องกันมากน้อยแค่ไหนและอย่างไร
2. คะแนนความนิยมของเพจ สามารถเพิ่มได้โดยการพยายามทำให้โปรแกรมค้นหามองว่าเพจตัวเองได้รับความนิยมสูง โดยการพยายามสร้างเพจของตัวเองขึ้นมาเพิ่มคะแนน หรือการพยายามทำให้เว็บเพจที่เป็นที่นิยมโหวตคะแนนให้แก่เว็บเพจของตนเอง ซึ่งเทคนิคที่ใช้นั้นจะกล่าวถึงโดยละเอียดในภายหลัง

เทคนิคและวิธีเหล่านี้กล่าวได้ว่าไม่เหมาะสม เพราะเป็นการพยายามทำให้โปรแกรมค้นหาเพิ่มคะแนนให้แก่เพจตัวเองโดยตรง โดยที่ไม่ได้มีการปรับปรุงคุณภาพของเว็บเพจ ซึ่งหากเรายอมรับให้มีการใช้เทคนิคเหล่านี้ในการเพิ่มคะแนนโดยไม่มีการพยายามตรวจจับหรือลงโทษ เพจที่อยู่ในลำดับต้นของผลค้นหาก็จะกลายเป็นเพจที่ใช้เทคนิคสแปมได้มีประสิทธิภาพที่สุด ซึ่งจะขัดแย้งกับความต้องการของผู้ใช้ที่ต้องการให้เพจที่มีคุณภาพสูงสุดที่ตรงกับคำค้นหา

เทคนิคที่ใช้ในการทำเว็บสแปมนั้นอาจแบ่งออกเป็น 2 ประเภทหลักได้คือ

1. คอนเทนท์สแปม (content spam) เป็นการเน้นไปที่การพยายามทำให้คำค้นหาที่เป็นที่นิยมนั้นมีความเกี่ยวข้องกับเพจของตน

2. ลิงก์สแปม (link spam) เป็นการเน้นไปที่การพยายามปรับเปลี่ยนโครงสร้างลิงก์ให้โปรแกรมค้นหามองว่าเว็บเพจของตนนั้นมีความนิยมสูง จะทำให้เว็บเพจนั้นได้รับลำดับสูงขึ้นในผลค้นคืน

ในงานวิจัยของเรานั้นเน้นไปที่การจับเว็บสแปมที่ใช้เทคนิคลิงก์สแปมเป็นหลัก เพราะการทำคอนเทนท์สแปมนั้นเป็นการพยายามนำเนื้อหาที่ไม่ตรงกับคำค้นหาไปให้ผู้ใช้ค้น ซึ่งทำให้เมื่อผู้ใช้เข้าไปดูเพจแล้วจะทราบได้ทันทีว่าเพจนี้เป็นเพจที่ตนเองไม่ต้องการ ตรงข้ามกับลิงก์สแปมที่เป็นการทำให้เพจที่เนื้อหาตรงกับคำค้นหาอยู่แล้วได้ลำดับในผลค้นคืนสูงขึ้น ดังนั้นจึงเป็นการยากที่ผู้ใช้จะมองเห็นว่าเพจใดบ้างที่ใช้เทคนิคลิงก์สแปมซึ่งสำหรับเพจทางการค้าแล้วเพจใดก็ตามที่อยู่ด้านบนของลำดับผลค้นคืนจะถูกผู้ใช้เข้าไปเยี่ยมชมก่อน ทำให้เพจที่ไม่ได้ใช้เทคนิคลิงก์สแปมนั้นเสียเปรียบเป็นอย่างมาก ดังนั้นลิงก์สแปมจึงเป็นปัญหาที่น่าสนใจกว่าเมื่อเทียบกับคอนเทนท์สแปม

ในงานวิจัยทางเว็บสแปมทั่วไป จะนิยามลิงก์สแปมไว้ว่า [16] เป็นโครงสร้างลิงก์ที่ถูกผู้ทำเว็บสแปมสร้างขึ้นโดยหวังว่าจะช่วยเพิ่มคะแนนความสำคัญให้แก่เว็บเพจของตน ดังนั้นในงานวิจัยของเราจะนิยามความหมายของเว็บสแปมว่าเป็นเว็บเพจที่ใช้มีการใช้เทคนิคลิงก์สแปมเพื่อเพิ่มคะแนนความสำคัญของเพจของตน

3.2 เทคนิคการสร้างลิงก์ฟาร์มและบูสต์เพจ

ลิงก์ฟาร์มนั้นเป็นหนึ่งในเทคนิคการทำลิงก์สแปม ซึ่งเป็นเทคนิคที่กำลังได้รับความนิยมสูงขึ้นมากในปัจจุบัน [16] อย่างที่ได้กล่าวไว้ในบทที่ 2.1.2 ว่าลิงก์ฟาร์มคือเซ็ตของเพจที่เชื่อมต่อกันอย่างหนาแน่น โดยมีเป้าหมายในการเร่งคะแนนเพจแรงก์ของเพจในลิงก์ฟาร์มของตนเอง เว็บเพจในลิงก์ฟาร์มแบ่งตามวัตถุประสงค์ในการสร้างอาจแบ่งได้เป็นสองประเภทคือ

1. เพจเป้าหมาย (target page) คือเพจที่ผู้สร้างลิงก์ฟาร์มต้องการทำการเร่งคะแนนให้ไปอยู่ในลำดับที่สูงของผลค้นคืน
2. บูสต์เพจ (boost page) คือเพจที่ผู้สร้างลิงก์ฟาร์มสร้างขึ้นเพื่อทำหน้าที่เร่งคะแนนให้แก่เพจเป้าหมาย

ในการสร้างลิงก์ฟาร์มให้เร่งคะแนนเพจเป้าหมายนั้นจำเป็นจะต้องมีการสร้างบูสต์เพจเป็นจำนวนมาก แล้วบังคับให้ชี้เข้าหาเพจที่เป็นเพจเป้าหมายหรือเพจในลิงก์ฟาร์มด้วยกัน โครงสร้างของเพจในลิงก์ฟาร์มมักจะประกอบด้วยเพจเป้าหมายจำนวนไม่มากนัก (ในหลายกรณีอาจจะมีแค่เพจเดียว) และบูสต์เพจอีกจำนวนหนึ่งที่มากพอที่จะเร่งคะแนนเพจแรงก์ได้อย่างมีประสิทธิภาพ

ในการสร้างเว็บสแปมนั้นผู้สร้างเว็บสแปมจะต้องหลีกเลี่ยงการตรวจจับเว็บสแปมของโปรแกรมค้นหา (search engine) ซึ่งผู้สร้างเว็บสแปมจะพยายามซ่อนลักษณะเนื้อหา โครงสร้างลิงก์ ของเว็บเพจเป้าหมายของตนเอง เพราะตราบใดที่เว็บเพจเป้าหมายของตนนั้นอยู่ในลำดับสูงในผลค้นคืน (search result) แล้ว การที่เพจบางเพจในลิงก์ฟาร์มจะถูกตรวจจับว่าเป็นสแปมก็ไม่ได้สร้างความเสียหายให้แก่เพจเป้าหมายมากนัก และนอกจากนั้นการสร้างบูสต์เพจยังต้องสร้างเป็นจำนวนมากถึงจะสามารถเร่งคะแนนเพจแรงก์อย่างมีนัยสำคัญได้ ดังนั้นบูสต์เพจเหล่านี้จึงมักถูกสร้างด้วยระบบอัตโนมัติ จึงเป็นการยากที่จะซ่อนลักษณะเฉพาะของบูสต์เพจได้ ดังนั้นในงานวิจัยนี้จึงเสนอแนวคิดที่ทำการตรวจจับเว็บสแปม โดยเริ่มต้นจากการตรวจจับบูสต์เพจ (boost page) แล้วจึงต่อยอดไปจนถึงตัวเพจที่เป็นเว็บสแปมจริง

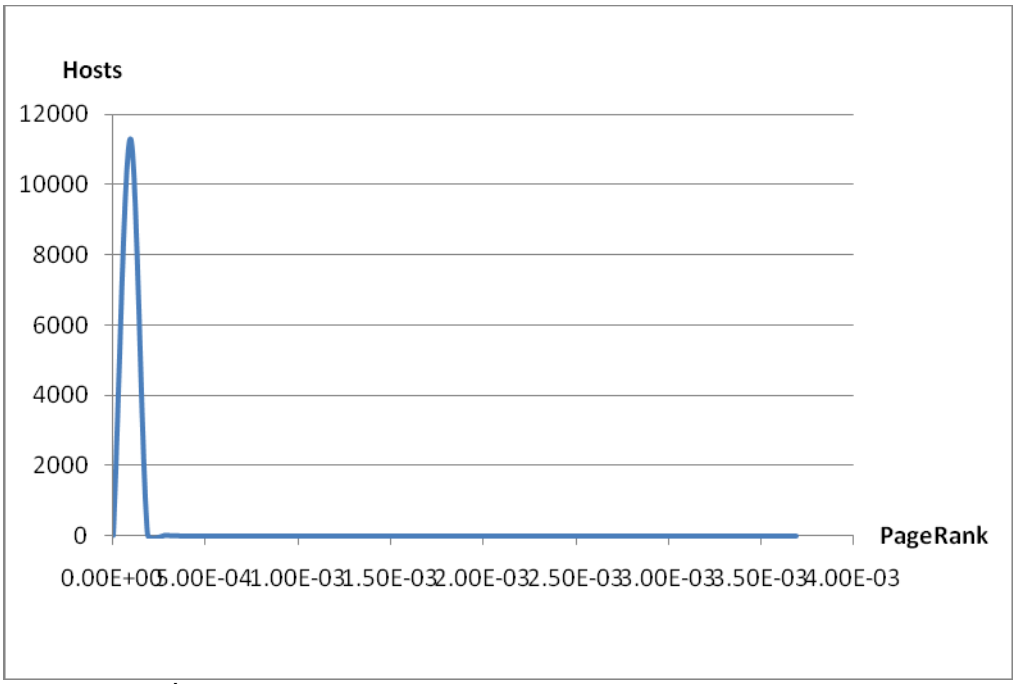
งานวิจัยนี้จึงแบ่งขั้นตอนวิธีการตรวจจับเว็บสแปมออกเป็นสองขั้นคือ

1. ขั้นตอนการตรวจจับบูสต์เพจ
2. ขั้นตอนการนำบูสต์เพจมาใช้ในการตรวจจับเว็บสแปม

เมื่อทำงานครบทั้งสองขั้นตอนเราก็จะได้ผลลัพธ์คือเว็บสแปมตามต้องการ

3.3 การกระจายตัวของคะแนนเพจแรงก์และการเร่งคะแนน

สำหรับการกระจายของคะแนนเพจแรงก์ในเว็บกราฟนั้นเป็นการกระจายแบบเพาเวอร์ลอว์ (power-law distribution) เมื่อนำไปพล็อตกราฟจะเป็นดังภาพที่ 3.1



ภาพที่ 3.1 กราฟแสดงการกระจายตัวของคะแนน PageRank ตามจำนวน Host

จากข้อมูลจะพบว่าคะแนนเพจแรงก์ของโฮสต์ในเว็บกราฟนั้นจะกระจุกตัวอยู่ที่บริเวณคะแนนเพจแรงก์ต่ำเป็นจำนวนมาก ซึ่งหากเราแบ่งช่วงคะแนนเพจแรงก์ออกมาเป็นตารางความถี่ตามตารางที่ 3.1

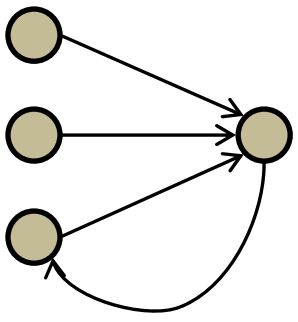
ช่วงคะแนน PageRank (10^{-4})	จำนวนโฮสโหนด
0.00 – 0.09	19
0.09 – 0.17	9508
0.18 – 0.27	898
0.28 – 0.36	367
0.37 – 0.45	188
0.46 – 0.54	114
0.55 – 0.64	72
0.65 – 0.73	54
0.74 – 0.82	33
0.83 – 0.91	36
0.92 – 1.01	25
1.02 – 1.10	14
1.11 – 1.19	6
1.20 – 1.28	4
1.29 – 1.37	6
1.38 – 1.47	6
1.48 – 1.56	6
1.57 – 1.65	3
1.66 – 1.74	2
1.75 – 1.84	3
1.85 – 1.93	6
1.94 – 2.02	2
2.03 – 2.11	2
2.12 – 2.21	0
2.22 – 2.30	4
2.31 – 2.39	5

ช่วงคะแนน PageRank (10^{-4})	จำนวนโฮสโหนด
2.40 – 2.48	3
2.49 – 2.58	1
2.59 – 2.67	0
2.68 – 2.76	0
2.77 – 2.85	1
2.86 – 2.94	1
2.95 – 3.04	1
.....	…..
3.05 – 6.00	4
6.01 – 36.93	8

ตารางที่ 3.1 จำนวนโหนดที่มีคะแนนเพจแรงก์ในช่วงต่างๆ

จากตารางที่ 3.1 จะเห็นว่าโดยส่วนมากแล้วโฮสทั่วไปจะมีคะแนนเพจแรงก์ไม่สูงมากนัก โดยตกอยู่ในช่วงคะแนนเดียวกันมากถึง 9508 โหนด คำถามที่น่าสนใจคือหากผู้สร้างสแปมต้องการจะเร่งคะแนนเพจแรงก์โดยการใช้ลิงก์ฟาร์ม เพื่อให้อย่างน้อยเพจของตนมีคะแนนเพจแรงก์เท่ากับเพจปกติทั่วไป จำเป็นจะต้องใช้บูสต์เพจจำนวนมากน้อยแค่ไหนอย่างไร

เพื่อค้นหาจำนวนบูสต์เพจที่จำเป็นต้องใช้เพื่อเร่งคะแนนเพจแรงก์ให้มากพอที่จะทำให้เพจเหล่านี้เป็นเพจที่มีคะแนนอยู่ในเพอร์เซ็นไทล์ที่ 20% บน เราทดลองสร้างลิงก์ฟาร์มตามแบบจำลองสแปมฟาร์มเหมาะสมที่สุด (optimal spam farm) ซึ่งจะมีลักษณะโครงสร้างดังเงื่อนไขต่อไปนี้ [10]



ภาพที่ 3.2 โครงสร้างลิงก์ของ Optimal Spam Farm

1. บูสต์เพจจะต้องชี้เข้าหาเพจเป้าหมายเพียงอย่างเดียวเท่านั้น

2. เพจเร่งจะต้องชี้หาบางบูสต์เพจ

โครงสร้างดังกล่าวเป็นไปตามภาพที่ 3.2 ซึ่งจะนำมาหาค่าคะแนนเพจแรงก์ที่ได้จากโครงสร้างดังกล่าวได้ด้วยขั้นตอนการคำนวนดังต่อไปนี้

<u>สมมติฐานในการคำนวณ</u>

1. เพจทั้งหมดในลิงก์ฟาร์มไม่ได้รับลิงก์จากเพจที่มีลักษณะเข้าถึงได้

2. เพจเป้าหมายชี้เข้าหาบูสต์เพจจำนวน 1 เพจ

<u>วิธีการคำนวณ</u>

กำหนดให้บูสต์เพจที่ไม่ได้รับลิงก์จากเพจเป้าหมายมีจำนวนทั้งหมด n เพจ

กำหนดให้เพจทั้งหมดในเว็บกราฟมีจำนวน N เพจ

เนื่องจากเพจเป้าหมายเหล่านี้ไม่ได้รับลิงก์เข้าเลย ดังนั้นคะแนนเพจแรงก์ของบูสต์เพจเหล่านี้จึงมีค่าเท่ากันในทุกๆ เพจ กำหนดให้คะแนนเพจแรงก์ของบูสต์เพจที่ไม่ได้รับลิงก์จากเพจเป้าหมายมีค่า P_b

กำหนดให้บูสต์เพจที่ได้รับลิงก์จากเพจเป้าหมายมีคะแนนเพจแรงก์ P_s

จากสมการคะแนนเพจแรงก์ปกติ

$$p_x = \alpha \sum_{(y,x)} \frac{p_y}{out(y)} + \frac{1-\alpha}{n} \qquad (3.1)$$

เมื่อนำมาคิดหา P_t จะแทนค่าตัวแปรได้ดังสมการต่อไปนี้

$$P_t = \alpha(P_b n + P_s) + \frac{(1-\alpha)}{N} \qquad (3.2)$$

โดยที่คะแนน P_b มีค่าเท่ากับ $(1-\alpha)/N$ และ P_s มีค่าดังสมการที่ 3.3

$$P_s = \alpha P_t + \frac{(1-\alpha)}{N} \quad (3.3)$$

เมื่อนำสมการที่ 3.2 และ 3.3 มารวมกันแล้วแทนค่าคะแนน P_b แล้วจะได้สมการหาคะแนน P_t ในเทอมของค่า n ได้ดังสมการที่ 3.4

$$P_t = \frac{(\alpha n + \alpha + 1)}{N(1+\alpha)} \quad (3.4)$$

โดยปกติแล้วค่า α จะนิยมใช้ที่ 0.85 ส่วน N หรือจำนวนเพจในเว็บกราฟของข้อมูลชุดทดสอบนั้นมีจำนวนทั้งหมด 77,741,046 เพจ ดังนั้นคะแนนเพจแรงก์ของเพจเป้าหมายจะมีค่าตามสมการ

$$P_t = \frac{0.85n + 1.85}{143821601.1} \quad (3.5)$$

จากตารางที่ 3.1 เมื่อเราพิจารณาว่าเว็บเพจถึง 83.4% มีคะแนนเพจแรงก์อยู่ในช่วงอันตรภาคชั้นที่ 0.09-0.17 x 10^{-4} ซึ่งหากผู้สร้างเว็บสแปม ต้องการเร่งคะแนนเพจแรงก์ให้อยู่ในอันตรภาคชั้นเดียวกับเพจส่วนมาก จำเป็นจะต้องใช้บูสต์เพจทั้งหมดมากถึง 1521 เพจ (คำนวณได้จากการแทนค่า P_t ด้วย 0.09×10^{-4})

ซึ่งสรุปได้ว่าในการเร่งคะแนนเพจแรงก์อย่างมีนัยสำคัญ แทบจะเป็นไปไม่ได้เลยที่จะใช้เพียงแค่โหนดในลิงก์ฟาร์มของตัวเอง จำเป็นต้องใช้โหนดที่มีลักษณะเข้าถึงได้ (accessible node) ช่วยในการเร่งคะแนนประกอบ ซึ่งข้อเท็จจริงนี้ทำให้ในการตรวจจับบูสต์เพจนั้นเราต้องพิจารณาทั้งในแง่ของบูสต์เพจที่เป็นเพจในลิงก์ฟาร์มของผู้สร้างสแปม และพิจารณาถึงบูสต์เพจที่เป็นเพจที่มีลักษณะเข้าถึงได้ไปพร้อมๆ กันด้วย เนื่องจากมีโอกาสน้อยมากที่เพจที่เป็นเว็บสแปมนั้นจะได้รับคะแนนจากลิงก์ฟาร์มเพียงอย่างเดียว ดังนั้นการพิจารณาความเป็นเว็บสแปมโดยอิงจากจำนวนเพจในลิงก์ฟาร์มเพียงอย่างเดียวจึงไม่เหมาะสม

3.4 การตรวจจับบูสต์เพจ

การแยกลักษณะเฉพาะของบูสต์เพจออกจากเพจอื่นๆ ในเว็บกราฟนั้นทำได้ไม่ง่ายนัก โดยเฉพาะเมื่อเราพิจารณาจากข้อเท็จจริงที่ว่าลิงก์ฟาร์มมีการเชื่อมต่อกัน ทำให้บูสต์เพจเพจหนึ่งอาจจะทำหน้าที่เร่งคะแนนให้แก่เพจเป้าหมายมากกว่าหนึ่งเพจได้

จากการพิจารณาข้อมูลในเว็บกราฟจริงเราพบว่าในสถานการณ์จริง ผู้สร้างสแปมไม่ได้สร้างลิงก์ฟาร์มให้เป็นไปตามแบบจำลองลิงก์ฟาร์มเหมาะสมที่สุด (optimal link farm) [10] ที่จะช่วยให้ผู้สร้างลิงก์ฟาร์มสามารถเร่งคะแนนให้แก่เพจเป้าหมายได้อย่างสูงสุด ซึ่งทั้งนี้เนื่องจากในสถานการณ์จริงแล้วการที่เว็บเพจของตนเองจะไม่มีลิงก์ออกไปยังหน้าอื่น ย่อมทำให้ผู้ใช้รู้สึกว่าเพจนั้นไม่มีความเป็นธรรมชาติ และไม่น่าสนใจ เราจะเห็นได้จากตารางที่ 3.2 ว่าโฮสที่ชี้ไปยังโฮสที่เป็นสแปมอย่างเดียวเมื่อเทียบกับเพจที่ชี้ไปยังเว็บปกติเพียงอย่างเดียว หรือชี้ไปยังโฮสที่เป็นทั้งโฮสปกติและโฮสสแปม จะมีจำนวนน้อยกว่าอย่างเห็นได้ชัด

ประเภท	จำนวนโฮส
โฮสที่ชี้ไปยังโฮสปกติเพียงอย่างเดียว	8954
โฮสที่ชี้ไปยังโฮสสแปมเพียงอย่างเดียว	646
โฮสที่ชี้ไปยังโฮสที่เป็นสแปมและโฮสปกติ	1802

ตารางที่ 3.2 จำนวนของโฮสที่ชี้ไปไปยังโฮสประเภทต่างๆ

ในการตรวจจับบูสต์เพจ เราจึงสรุปได้ว่าเป็นไปได้ที่บูสต์เพจนั้นจะชี้ไปยังเพจที่เป็นเพจธรรมดาด้วย แต่แน่นอนว่าบูสต์เพจย่อมชี้ไปยังเพจที่เป็นสแปม มากกว่าเพจที่เป็นเพจธรรมดา ดังนั้นเราจึงกำหนดนิยามของบูสต์เพจไว้ดังนี้

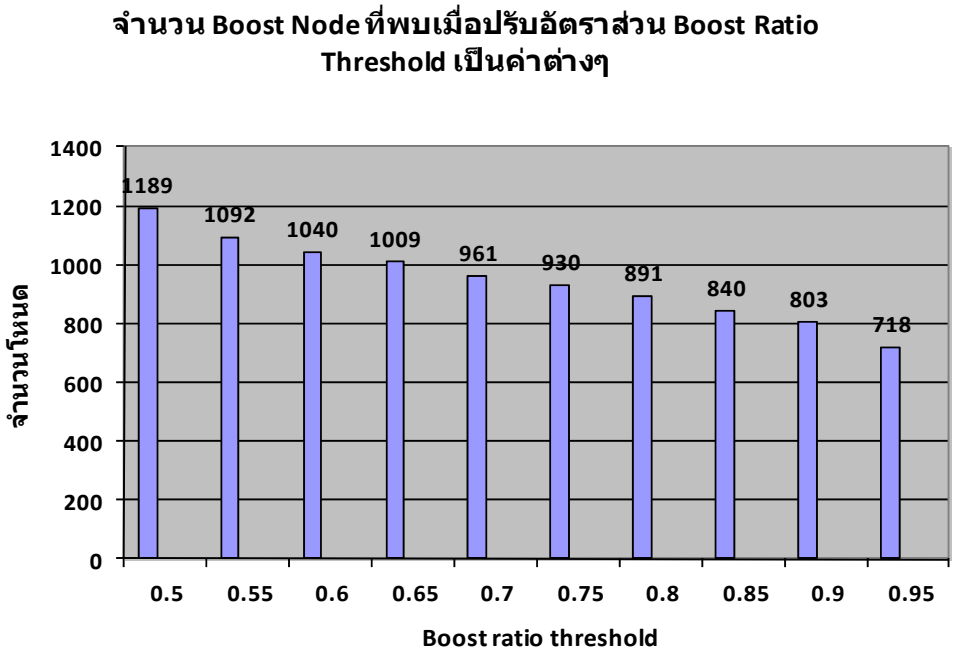
นิยามที่ 3.1 *กำหนดให้ S_x เป็นจำนวนลิงก์ในโหนด x ที่ชี้ไปยังโหนดที่เป็นโหนดสแปม และโหนด N_x เป็นจำนวนลิงก์ในโหนด x ที่ชี้ไปยังโหนดที่เป็นโหนดปกติ เมื่อกำหนดค่า***อัตราส่วนเริ่มต้นการเป็นโหนดเร่ง** (boosting ratio threshold) *t แล้ว โหนด x ใดๆ ในเว็บกราฟ เป็นบูสต์โหนด (boost node) ก็ต่อเมื่อ*

$$\frac{S_x}{S_x + N_x} \leq t \tag{3.6}$$

นิยามดังกล่าวนั้นเกิดจากแนวคิดที่ว่าบูสต์เพจนั้นเป็นเพจที่ลิงก์ออกโดยส่วนมากจะชี้ไปยังเพจที่เป็นสแปม โดยที่มีค่าอัตราส่วนเริ่มต้นการเป็นโหนดเร่ง เป็นตัวแปรที่ผู้จับสแปมสามารถกำหนดความละเอียดในการตรวจจับได้ หากตั้งค่าอัตราส่วนเริ่มต้นการเป็นโหนดเร่งไว้สูงจะทำให้มีโอกาสเกิดความคลาดเคลื่อนประเภทที่ 1 (type 1 error) ค่อนข้างสูง และโอกาสเกิดความ

คลาดเคลื่อนประเภทที่ 2 (type 2 error) ต่ำ ในทางตรงกันข้ามหากเราตั้งค่าอัตราส่วนการเป็นโหนดเร่งไว้ต่ำ ย่อมทำให้มีโอกาสเกิดความคลาดเคลื่อนประเภทที่ 1 ต่ำ แต่ก็มีโอกาสเกิดความคลาดเคลื่อนประเภทที่ 2 ค่อนข้างสูง

เราทดลองนำนิยามที่ 3.1 ไปทดสอบเพื่อค้นหาบูสต์โหนด โดยทำการทดลองบนชุดข้อมูลทดสอบ เราพบว่าจำนวนโหนดเร่งที่ค้นพบเมื่อปรับค่าอัตราส่วนการเป็นโหนดเร่งเป็นไปดังภาพที่ 3.3



ภาพที่ 3.3 แผนภาพแสดงจำนวนบูสต์โหนดที่พบในอัตราส่วนการเป็นโหนดเร่งต่างๆ

เมื่อลองปรับอัตราส่วนโหนดเริ่มต้นการเป็นโหนดเร่งให้มากขึ้นจะพบว่าจำนวนบูสต์โหนดนั้นจะลดลงไม่มากนัก ข้อมูลนี้ช่วยสนับสนุนข้อเท็จจริงที่ว่าโหนดปกติมักจะไม่ชี้เข้าหาโหนดที่เป็นสแปม เนื่องจากตัวเลขที่ลดลงโดยเฉลี่ย 50 โหนด ต่ออัตราส่วนที่เปลี่ยนไป 0.05 แปลว่ามีโหนดจำนวนไม่มากนักที่ชี้เข้าหาทั้งโหนดที่เป็นสแปมและโหนดที่เป็นโหนดธรรมดาในอัตราส่วนครึ่งต่อครึ่ง แปลว่าโหนดที่เป็นโหนดธรรมดาที่แท้จริงก็มักจะชี้เข้าหาเพียงโหนดธรรมดาและบูสต์โหนดก็มักจะชี้เข้าหาเพียงบูสต์โหนดเท่านั้น

จุดแข็งของนิยามที่เราใช้คือนิยามดังกล่าวสามารถตรวจจับลิงก์ฟาร์มได้ไม่ว่าลิงก์ฟาร์มดังกล่าวจะปรับโครงสร้างเป็นอย่างไร โดยปกติแล้วผู้สร้างสแปมย่อมพยายามจะเปลี่ยนลักษณะโครงสร้างของเว็บสแปมเพื่อหลีกเลี่ยงการถูกตรวจจับ แต่ไม่ว่าผู้สร้างสแปมจะพยายาม

ปรับเปลี่ยนดัดแปลงโครงสร้างของลิงก์ฟาร์มให้ซับซ้อนมากแค่ไหนก็ตาม การสร้างเว็บสแปมก็ยังจำเป็นต้องมีบูสต์เพจและมีเพจที่ชี้เข้าหาเสมอ ดังนั้นการที่เล็งเป้าไปที่เว็บเพจที่ชี้เข้าหาเว็บสแปมโดยตรงย่อมทำให้ผู้สร้างสแปมไม่สามารถหลีกเลี่ยงวิธีการตรวจจับดังกล่าวได้ นอกเสียจากผู้สร้างเว็บสแปมจะพยายามทำให้ลิงก์ที่ชี้เข้าหาเพจตัวเองลดลง ซึ่งการกระทำเช่นนั้นก็ย่อมเป็นผลไม่ดีแก่ตัวผู้สร้างเว็บสแปมเองเนื่องจากจะทำให้ค่าคะแนนเพจแรงก์มีค่าลดลงซึ่งเป็นการลงโทษไปในตัวอยู่แล้ว

ดังที่กล่าวไปในหัวข้อที่ 3.3 ว่าในการพิจารณาบูสต์เพจนั้นต้องพิจารณาทั้งบูสต์เพจที่อยู่ในลิงก์ฟาร์มของผู้สร้างสแปมเอง และบูสต์เพจที่มีลักษณะเข้าถึงได้ด้วย ซึ่งวิธีการตรวจจับโดยใช้ค่าอัตราส่วนเริ่มต้นการเป็นโหนดเร่งนั้น จะทำให้เราตรวจจับเพจที่เป็นเพจที่มีลักษณะเข้าถึงได้ที่มักถูกใช้เป็นตัวเร่งคะแนนให้แก่เว็บสแปมด้วย เนื่องจากเพจที่มีลักษณะเข้าถึงได้เหล่านี้จะถูกใช้เป็นแหล่งคะแนนของเว็บสแปมจำนวนมาก ดังนั้นเพจเหล่านี้จะถูกฝังลิงก์ที่ชี้เข้าหาเพจสแปม ซึ่งหากลิงก์ที่ถูกฝังนั้นมีจำนวนมากจนเกินอัตราส่วนเริ่มต้นการเป็นโหนดเร่ง เราก็จะค้นพบโหนดเร่งเหล่านี้ด้วยนั่นเอง

นอกจากนี้งานวิจัยที่ [13] ยังได้กล่าวไว้อย่างชัดเจนว่าในสถานการณ์จริงแล้วผู้ทำเว็บสแปมมักจะมีการร่วมมือกันเป็นพันธมิตรลิงก์สแปม (link spam alliance) เนื่องจากว่าเมื่อเทียบกันผู้สร้างเว็บสแปมที่นำลิงก์ฟาร์มของตนเองเข้ามาเชื่อมต่อกับผู้สร้างเว็บสแปมคนอื่นเพื่อเร่งคะแนนให้เว็บสแปมของแต่ละคน จะได้รับคะแนนเพจแรงก์สูงกว่าผู้สร้างเว็บสแปมที่อาศัยลิงก์ฟาร์มของตัวเองเพียงอย่างเดียว ดังนั้นในบูสต์เพจจำนวนหนึ่งจึงมีลิงก์ออกชี้ไปยังเพจที่เป็นเพจเป้าหมายมากกว่าหนึ่งเพจ ซึ่งในทางกลับกันแล้วสามารถกล่าวได้ว่าถึงแม้ว่าเราอาจจะมีข้อมูลเว็บสแปมที่ไม่สมบูรณ์ทั้งหมด เราก็ยังสามารถตรวจจับบูสต์เพจได้อย่างมีประสิทธิภาพ ซึ่งคุณลักษณะดังกล่าวทำให้เราสามารถใช้ข้อมูลเว็บสแปมเพียงบางส่วนในการขยายผลไปหาเพจเร่งคแนน แล้วนำไปใช้ในการตรวจสอบเว็บสแปมทั้งหมด อาจสรุปได้ว่าโดยการใช้นิยามที่ 3.1 นั้นเราสามารถตรวจจับเว็บสแปมจำนวนมากได้โดยการป้อนข้อมูลเว็บสแปมที่เรารู้ล่วงหน้าเพียงบางส่วนเท่านั้น ซึ่งผลการทดลองในบทที่ 4 จะกล่าวถึงประเด็นนี้อีกรอบ

ข้อมูลที่ได้จากการค้นหาบูสต์โหนดจะสามารถนำไปใช้ในการค้นหาเว็บสแปมต่อไป

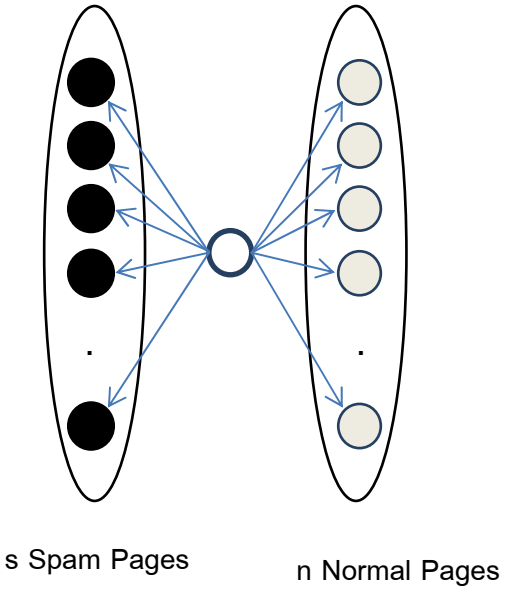
3.5 การตรวจจับเว็บสแปมโดยอาศัยบูสต์โหนด

เนื่องจากในขั้นตอนที่ผ่านมาเราได้ทำการตรวจจับบูสต์โหนดไปแล้ว ในขั้นตอนการตรวจจับเว็บสแปมโดยอาศัยบูสต์โหนดที่ได้มานั้นจะตั้งอยู่ในแนวคิด 2 อย่างคือ

1. บูสต์โหนดนั้นได้มาจากการป้อนข้อมูลเว็บสแปมเพียงบางส่วนเท่านั้น

2. ผู้ทำเว็บสแปมจะพยายามทำให้บูสต์โหนดชี้เข้าหาเพียงเฉพาะเว็บสแปมเท่านั้น

แนวคิดที่หนึ่งนั้นมีจุดเริ่มต้นคือการค้นหาบูสต์โหนดด้วยนิยามที่ 3.1 นั้นจำเป็นจะต้องใช้ข้อมูลเว็บสแปมป้อนเข้าไปก่อนเพื่อค้นหาโหนดที่มีลักษณะตรงกับนิยาม ดังนั้นผู้ตรวจจับเว็บสแปมจำเป็นจะต้องเก็บข้อมูลเว็บสแปมเสียก่อนจึงจะสามารถค้นหาบูสต์โหนดได้ แต่ในทางกลับกันเนื่องจากเรากำลังทำการตรวจจับเว็บสแปม ผู้ตรวจจับเว็บสแปมต้องการความรู้ที่ว่าเว็บใดเป็นเว็บสแปมบ้าง ผู้ตรวจจับสแปมเป็นผู้ที่ต้องการข้อมูลว่าโหนดใดบ้างเป็นสแปม ดังนั้นแล้วการที่ผู้ตรวจจับสแปมจะมีข้อมูลนี้อยู่แล้วและป้อนลงไปในนิยามที่ 3.1 ตั้งแต่ต้นจึงเป็นสถานการณ์ที่ไม่สมควรเกิดขึ้น ดังนั้นสิ่งที่เป็นไปได้มากที่สุดคือผู้ตรวจจับสแปมนั้นมีข้อมูลของเว็บสแปมเพียงบางส่วนและต้องการค้นหาเว็บสแปมส่วนใหญ่ให้ได้



ภาพที่ 3.4 ลิงก์ออกของบูสเพจแบ่งเมื่อแบ่งออกเป็นสองจำพวก

เหตุผลในการตั้งแนวคิดที่สองนั้นสามารถอธิบายได้ด้วยสมการเพจแรงก์ ซึ่งอธิบายได้ดังนี้ กำหนดให้บูสต์เพจใดๆ B มีลิงก์ชี้เข้าหาเซตของเพจเป้าหมายซึ่งเป็นเว็บสแปม $\{T_1, T_2, T_3, \ldots, T_s\}$ จำนวน s ลิงก์ และมีลิงก์ชี้เข้าหาเซตของเพจปกติ $\{I_1, I_2, I_3, \ldots, I_n\}$ จำนวน n ลิงก์ ตามภาพที่ 3.4 คะแนนเพจแรงก์ของโหนด T แต่ละโหนดที่ได้รับจากโหนด B นั้นเป็นไปตามสมการที่ 3.7

$$P_T = 0.85 \frac{P_B}{s + n} \quad (3.7)$$

จากสมการจะพบว่าจำนวนลิงก์ออกไปหาเพจธรรมดาจะแปรผกผันกับคะแนนเพจแรงก์ที่ได้รับจากบูสต์เพจ ดังนั้นผู้สร้างบูสต์เพจจึงต้องพยายามหลีกเลี่ยงการสร้างลิงก์ชี้ออกไปยังเพจธรรมดาให้มากที่สุดเท่าที่จะทำได้

จากแนวคิดสองข้อดังกล่าวทำให้เรานิยามโหนดที่เป็นสแปมจากข้อมูลของบูสต์โหนดได้ดังนิยามต่อไปนี้

นิยามที่ 3.2 *กำหนดเซตของบูสต์โหนด* $B = \{b_1, b_2, b_3, \ldots, b_n\}$ *โหนด* S *ใดๆ ในเว็บกราฟจะเป็นโหนดสแปมก็ต่อเมื่อ มีอย่างน้อยหนึ่งลิงก์เป็นลิงก์ที่บางโหนด* $b \in B$ *ชี้เข้า* S

นิยามดังกล่าวนั้นสอดคล้องกับแนวคิดข้อหนึ่งและข้อสอง เนื่องจากว่าในแนวคิดแรกนั้นกล่าวว่าบูสต์โหนดที่ได้มานั้นเกิดจากการย้อนลิงก์ตามนิยามที่ 3.1 โดยใช้ข้อมูลเว็บสแปมเพียงแค่บางส่วน แต่บูสต์โหนดจะพยายามชี้เข้าหาโหนดสแปมให้มากที่สุดเท่าที่จะมากได้ ดังนั้นแล้วเราจึงกำหนดว่าลิงก์ที่ไม่ได้เกิดจากการย้อนกลับตามนิยามที่ 3.1 ก็ย่อมเป็นลิงก์ที่ชี้ออกไปยังสแปมเช่นกัน

นอกจากนี้นิยามดังกล่าวยังทนทานต่อการเปลี่ยนแปลงโครงสร้างของเว็บสแปมด้วย กล่าวคือไม่ว่าผู้สร้างเว็บสแปมจะพยายามเปลี่ยนแปลงโครงสร้างของบูสต์เพจให้มีการชี้ไปยังเพจธรรมดา ก็ไม่สามารถหลีกเลี่ยงการถูกตรวจจับด้วยนิยามนี้ได้ เนื่องจากทุกลิงก์ที่ชี้ออกจากบูสต์เพจจะถูกบังคับให้เป็นเพจสแปมทั้งหมด ดังนั้นแล้วการเปลี่ยนแปลงโครงสร้างจึงทำให้เพจธรรมดาถูกตรวจจับเป็นเพจสแปมเท่านั้น แต่ไม่สามารถทำให้เพจของผู้สร้างสแปมหลีกหนีการตรวจจับไปได้

3.6 การตรวจสอบโหนดปกติ

จากขั้นตอนการตรวจจับโหนดที่เป็นสแปม เราจะพบว่านิยามที่ 3.2 นั้นยังมีข้อเสียอยู่ตรงที่ว่าทุกโหนดที่บูสต์เพจชี้ไปจะนับเป็นเพจสแปมทั้งหมด ซึ่งถึงแม้จะได้อธิบายไปแล้วว่าในสมการเพจแรงก์นั้นผู้ทำสแปมจะพยายามหลีกเลี่ยงการนำบูสต์เพจชี้ไปยังเพจปกติ แต่ในความเป็นจริงแล้ว เพจแร่งคะแนนที่ได้จากนิยามที่ 3.1 นั้นไม่ใช่บูสต์เพจที่อยู่ภายใต้การควบคุมของ

ผู้สร้างสแปมทั้งหมด ในการที่จะขยายผลไปยังเพจที่เป็นสแปมในเว็บกราฟโดยเริ่มจากข้อมูลสแปมเพียงเล็กน้อย เราจำเป็นต้องใช้บูสต์เพจที่อยู่ในกลุ่มเพจที่เข้าถึงได้ด้วย ซึ่งเพจที่เข้าถึงได้เหล่านี้ มีโอกาสสูงที่เป็นเพจปกติที่ผู้ใช้สแปมฝังลิงก์เข้าไปได้ ซึ่งส่วนมากเป็นเพจที่อยู่ในลักษณะของเว็บบอร์ดและบล็อก ซึ่งเนื่องจากเพจเหล่านี้ไม่ได้ต้องการเพิ่มคะแนนให้เว็บสแปมโดยเฉพาะ และเพจเหล่านี้เป็นเพจปกติที่ต้องเชื่อมโยงกับเพจอื่นๆ เป็นธรรมดา ทำให้เพจเหล่านี้มักจะชี้ไปหาเพจปกติด้วย ซึ่งถึงหากจะกำหนดค่าอัตราส่วนการเป็นโหนดเร่งไว้ให้สูงขึ้น เพื่อลดจำนวนโหนดเร่งที่ชี้ไปยังโหนดปกติ ก็จะทำให้จำนวนโหนดสแปมที่ตรวจจับได้ลดลงเป็นอย่างมาก ซึ่งผลลัพธ์ความเปลี่ยนแปลงแสดงอยู่ในผลการทดลองในบทที่ 4

แนวคิดในการแก้ปัญหานี้ทำได้โดยการพยายามค้นหา*โหนดปกติอย่างหนักแน่น (firmly normal node)* ซึ่งเป็นกลุ่มของโหนดที่เรามั่นใจได้ว่าเป็นโหนดปกติ โหนดในกลุ่มนี้สามารถใช้ในการตรวจสอบซ้ำว่าโหนดที่เป็นสแปมตามนิยามที่ 3.2 นั้นโหนดใดบ้างที่มีโอกาสสูงที่จะเป็นโหนดปกติ ดังนั้นโหนดปกติอย่างหนักแน่นนี้สามารถใช้ในการช่วยลดความผิดพลาดในการตรวจจับโหนดปกติผิดพลาดเป็นโหนดสแปมได้

เซ็ตของโหนดปกติอย่างหนักแน่นนี้จะต้องมีความคลาดเคลื่อนประเภทที่ 2 (Type II error) น้อยกว่าการตรวจจับสแปม ทั้งนี้เพราะเซ็ตของโหนดปกติอย่างหนักแน่นนี้ใช้ในการตรวจสอบโหนดที่เป็นสแปมที่ได้จากนิยาม 3.2 ซ้ำอีกครั้ง ดังนั้นหากในกระบวนการค้นหาโหนดปกติอย่างหนักแน่นมีความคลาดเคลื่อนประเภทที่ 2 หรือกล่าวคือมีโหนดที่เป็นสแปมอยู่ในกลุ่มของโหนดปกติอย่างหนักแน่นสูง พอนำมาตรวจซ้ำแล้วจะทำให้โหนดที่เดิมทีนิยามที่ 3.2 กล่าวไว้ว่าเป็นโหนดสแปมนั้นหลุดรอดไปเป็นโหนดธรรมดาเป็นจำนวนมาก ทำให้การตรวจซ้ำนี้ลดประสิทธิภาพของระบบตรวจจับสแปมโดยรวมให้มีความแม่นยำน้อยลง แต่สามารถยอมรับความผิดพลาดประเภท 1 (Type I error) ได้สูงกว่า หรือกล่าวคือสามารถค้นหาโหนดปกติไม่พบส่วนหนึ่งได้ เนื่องจากว่าสิ่งที่สำคัญคือเรานำโหนดปกติเหล่านี้มาตรวจซ้ำ ซึ่งส่วนที่สามารถตรวจซ้ำได้จะมีเพียงโหนดที่พบทั้งในเซ็ตของโหนดที่เป็นสแปมและเซ็ตของโหนดปกติอย่างหนักแน่นพร้อมกันเท่านั้น ดังนั้นประเด็นสำคัญจึงไม่ใช่จำนวนของโหนดปกติที่พบในเว็บกราฟ แต่เป็นจำนวนของโหนดปกติที่สามารถนำไปตรวจสอบซ้ำกับเซ็ตของโหนดที่เป็นสแปมได้

สำหรับการค้นหาโหนดปกติอย่างหนักแน่น ทำได้โดยอาศัยหลักการว่าเว็บปกติย่อมจะหลีกเลี่ยงการชี้เข้าหาเว็บสแปม ซึ่งเป็นหลักการเดียวกันกับที่ใช้ในงานวิจัยทรัสแรงก์ (TrustRank) [6] ซึ่งทำการเรียงลำดับคะแนนความน่าเชื่อถือของเพจโดยเริ่มต้นจากเซ็ตของเพจที่เป็นมั่นใจว่าเป็นเพจปกติจากการตรวจสอบโดยผู้เชี่ยวชาญโดยตรง แล้วจึงนำเซ็ตที่มั่นใจว่าเป็นเพจปกติ

เหล่านี้ไปขยายผลค้นหาเพจที่เป็นเพจปกติ ด้วยหลักการเดียวกันนี้ การค้นหาโหนดปกติสามารถทำได้โดยอาศัยนิยามต่อไปนี้

นิยามที่ 3.3 *บนกราฟถ่วงน้ำหนัก G = (V, E) กำหนดเซ็ตของโหนดที่มั่นใจว่าเป็นโหนดปกติ* $N = \{n_1, n_2, n_3, ...\} \subset V$ *โหนด x ใดๆ เป็นโหนดปกติอย่างหนักแน่น ก็ต่อเมื่อ* $x \in N$ *หรือ มีบางโหนด y ที่มีลิงก์ออกที่มีน้ำหนักสูงสุด k ตัว ชี้ไปยังโหนด x และ* $y \in N$

นิยามที่ 3.3 นี้ได้กำหนดชัดเจนว่าใช้เฉพาะกับกราฟถ่วงน้ำหนักเท่านั้น ซึ่งหมายความว่านิยามนี้ต่างกับนิยามอื่นตรงที่สามารถใช้ได้เฉพาะบนโฮสกราฟเท่านั้น โฮสกราฟนั้นจะเป็นกราฟถ่วงน้ำหนักซึ่งน้ำหนักของเส้นเชื่อมหมายความถึงจำนวนลิงก์ที่เชื่อมกันระหว่างโฮสต์นั่นเอง

นิยาม 3.3 นี้นิยามขึ้นด้วยเหตุผลว่า โฮสปกติย่อมหลีกเลี่ยงที่จะให้มีลิงก์ภายในโฮสของตนเองชี้ไปยังเพจที่เป็นเว็บสแปม หมายความว่าเพจทั้งหมดที่อยู่ในการควบคุมของผู้สร้างเว็บเพจซึ่งเป็นเจ้าของโฮส ย่อมปราศจากลิงก์ออกไปหายังเว็บสแปม แต่ปัจจุบันนั้น โฮสมากมายจะมีเพจที่เป็นลักษณะตอบโต้กับผู้ใช้งานได้ (interactivable) ซึ่งผู้ใช้งานมีส่วนในการกำหนดเนื้อหาในเว็บเพจ ดังนั้นเพจในโฮสเหล่านี้มีความเป็นไปได้ที่ผู้สร้างสแปมจะฝังลิงก์ออกไปยังเพจที่เป็นสแปมของตนเอง เช่น เพจในภาพที่ 3.5

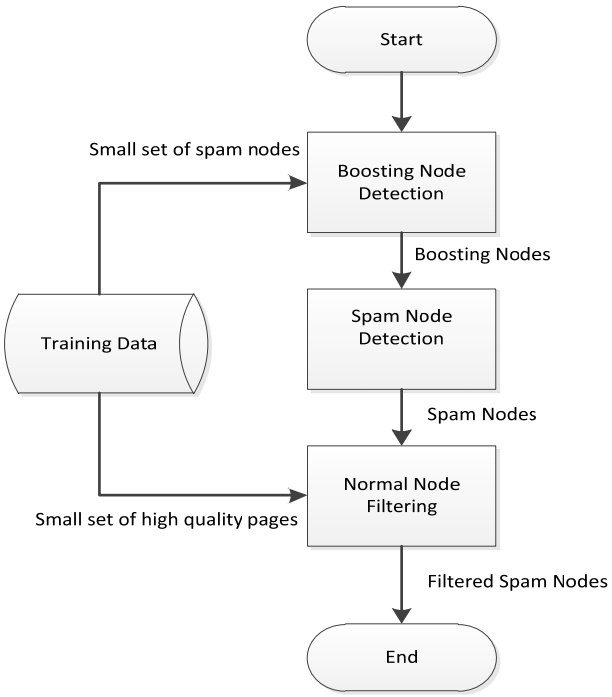
ภาพที่ 3.5 ตัวอย่างผู้สร้างสแปมฝังลิงก์ออกไปยังเพจสแปมของตนบนโฮสที่ไม่เป็นสแปม

แต่เนื่องจากว่าการฝังลิงก์ในลักษณะนี้ไม่สามารถทำได้บ่อยเนื่องจากเจ้าของเว็บเพจปกติที่มีคุณภาพจะมีการกวาดล้างและป้องกันไม่ให้มีลิงก์สแปมที่มักเป็นการโฆษณาสินค้า ฝังอยู่ในเว็บบอร์ดของตน อยู่อย่างสม่ำเสมอ ลิงก์ที่ชี้ออกจากเว็บเพจปกติที่มีคุณภาพสูงไปยังเว็บสแปมจึงมีจำนวนน้อยมากในช่วงระยะเวลาหนึ่ง ดังนั้นเราจึงสามารถป้องกันมิให้ลิงก์ที่ถูกฝังอยู่ในเพจปกติในลักษณะนี้ได้รับผลประโยชน์จากวิธีการค้นหาโหนดปกติอย่างหนักแน่น โดยกำหนดให้ลิงก์ที่ออกจากเว็บเพจปกติเป็นจำนวนสูงสุดเพียง k ตัว เป็นลิงก์ที่ปลอดภัยเท่านั้น โดยไม่สนใจลิงก์

ออกที่มีลักษณะชั่วคราว (temporary) และมีจำนวนน้อยเมื่อเทียบกับลิงก์อื่นๆ ในเว็บเพจนั้น จากผลการในบทที่ 4 เพื่อทดสอบประสิทธิภาพของนิยามแล้ว พบว่านิยามนี้สามารถตรวจสอบค้นหาโหนดที่เป็นปกติได้แม่นยำสูงขึ้นจากการสุ่มหาโหนดเป็นอย่างมาก จึงสามารถช่วยตัดความคลาดเคลื่อนประเภทที่บวกลวง (false-positive) จากการตรวจจับนิยามที่ 3.1 และ 3.2 ได้เป็นอย่างดี

3.7 สรุปอัลกอริธึมการตรวจจับลิงก์สแปม

การตรวจจับลิงก์สแปมเราจะใช้นิยามที่ 3.1, 3.2 และ 3.3 ประกอบเข้าด้วยกัน โดยภาพรวมของระบบทั้งหมดสามารถแสดงได้ตามภาพที่ 3.6 ดังนี้



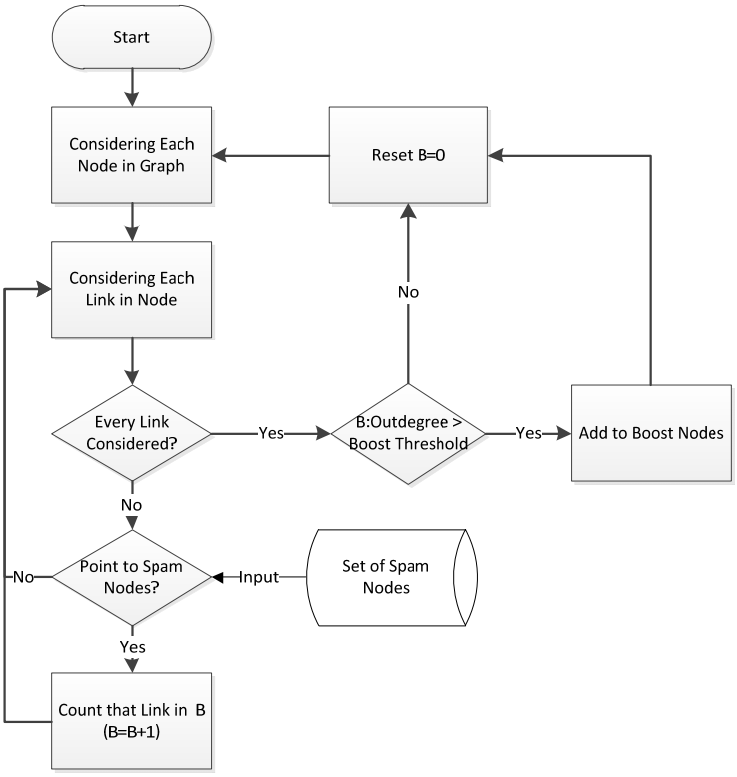
ภาพที่ 3.6 ผังงานแสดงขั้นตอนการทำงานโดยรวมของระบบ

อัลกอริธึมทั้งหมดนี้ต้องการข้อมูลชุดสอนเป็นเซตของโหนดที่เป็นสแปม และเซ็ตของโหนดที่เป็นโฮสคุณภาพสูง ซึ่งในการทดลองนั้นเซ็ตของโหนดที่เป็นสแปมจะใช้ชุดเดียวกับชุดข้อมูลสอนที่อยู่ในชุดข้อมูลเทียบสมรรถนะ (benchmark data) เพื่อให้สามารถเปรียบเทียบได้กับอัลกอริธึมอื่น และชุดข้อมูลโฮสคุณภาพสูงจะใช้เพจที่มีค่าคะแนนเพจแรงก์สูงและไม่เป็นเว็บสแปม

ซึ่งในแต่ละขั้นตอนการทำงานของระบบยังสามารถแบ่งละเอียดลงไปได้ดังต่อไปนี้

3.7.1 อัลกอริธึมการตรวจจับบูสต์โหนด

วิธีการตรวจจับบูสต์โหนดสามารถแสดงได้ตามผังงานตามภาพที่ 3.7



ภาพที่ 3.7 ผังงานแสดงอัลกอริธึมการตรวจจับบูสต์โหนด

ขั้นตอนการประมวลผลจะเริ่มจากพิจารณาโหนดต่างๆ ในเว็บกราฟ หากมีลิงก์ที่ชี้เข้าหาข้อมูลโหนดสแปม ให้ทำการนับจำนวนลิงก์เหล่านั้น แล้วนำมาใช้คำนวนว่าจำนวนลิงก์เหล่านี้มีอัตราส่วนเกินอัตราส่วนเริ่มต้นการเป็นโหนดเร่ง (boost ratio threshold) หรือไม่ หากมากกว่า ให้บอกว่าโหนดนั้นเป็นบูสต์โหนด

<u>วิเคราะห์ความซับซ้อน</u>

กำหนดให้

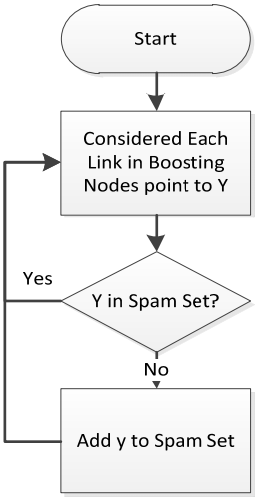
- จำนวนโหนดในเซ็ตของโหนดที่เป็นสแปม มีทั้งหมด n โหนด

- จำนวนลิงก์ทั้งหมดในเว็บกราฟ m ลิงก์

ในการพิจารณาหาว่าแต่ละโหนดเป็นบูสต์โหนดหรือไม่ จะต้องเทียบลิงก์ของแต่ละโหนดว่าชี้เข้าหาโหนดในเซ็ตที่เป็นสแปมหรือไม่ ซึ่งในกรณีที่ไม่มีการทำดัชนีในข้อมูลเซ็ตสแปม เราก็จะต้องเปรียบเทียบเป็นจำนวนสูงสุด n ครั้งต่อ 1 ลิงก์ จำนวนลิงก์ที่มีทั้งหมดในเว็บกราฟเป็น m ลิงก์ ดังนั้นความซับซ้อนของอัลกอริธึมจึงสามารถสรุปให้อยู่ในรูป O(mn)

3.7.2 อัลกอรธึมการตรวจจับโหนดที่เป็นสแปม

หลังจากที่เราได้เซ็ตของโหนดที่เป็นบูสต์โหนดแล้ว การนำข้อมูลดังกล่าวมาตรวจจับโหนดที่เป็นสแปม ด้วยนิยาม 3.2 สามารถทำได้ด้วยอัลกอริธึมตามผังงานในภาพที่ 3.8



ภาพที่ 3.8 **อัลกอริธึมการตรวจจับเว็บสแปมด้วยบูสต์โหนด**

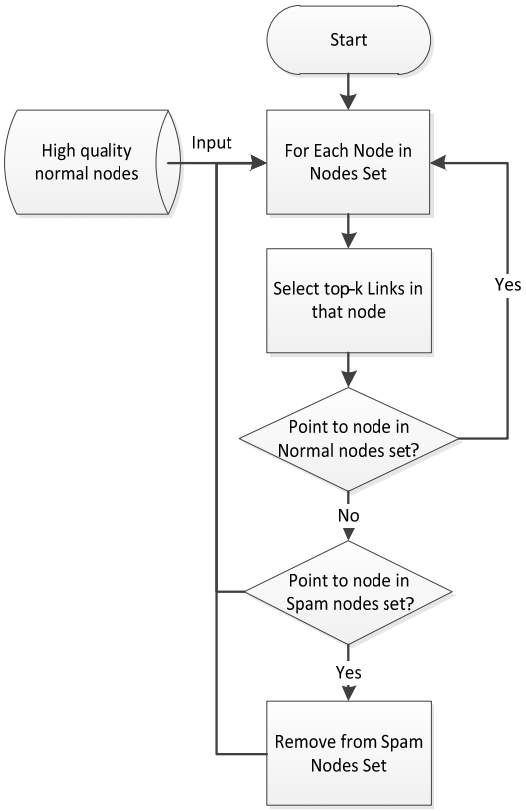
ขั้นตอนการประมวลผลนั้นเริ่มจากการดูลิงก์ทั้งหมดในเซ็ตของโหนดที่เป็นบูสต์โหนด แล้วบอกว่าโหนดที่ได้รับลิงก์ชี้เข้าจากบูสต์โหนดทุกโหนดเป็นโหนดสแปมนั่นเอง

<u>วิเคราะห์ความซับซ้อน</u>

เมื่อกำหนดให้จำนวนลิงก์ทั้งหมดที่บูสต์โหนดชี้มีจำนวน m โหนด เวลาที่ใช้ในการพิจารณาโหนดที่เป็นสแปมจะใช้ทั้งหมด O(m)

3.7.3 อัลกอริธึมการคัดกรองโหนดปกติ

การคัดกรองโหนดปกตินั้นทำได้โดยการใช้นิยามที่ 3.3 โดยอัลกอริธึมดังภาพที่ 3.9



ภาพที่ 3.9 ผังงานแสดงอัลกอริธึมการคัดกรองโหนดปกติ

ขั้นตอนการประมวลผลจะเริ่มจากการนำข้อมูลโหนดปกติคุณภาพสูงเข้ามาพิจารณาทีละโหนด แล้วเลือกมาเฉพาะลิงก์ที่มีจำนวนสูงสุด k ตัวเท่านั้น หากลิงก์เหล่านั้นชี้ออกไปยังโหนดที่เคยระบุว่าสแปมโหนด ให้กรองสแปมโหนดนั้นออกจากสแปมโหนดโดยการติดป้ายว่าเป็นโหนดปกติ และทำซ้ำเช่นนี้ไปเรื่อยๆ จนครบทุกโหนดในกลุ่มโหนดปกติคุณภาพสูง

<u>วิเคราะห์ความซับซ้อน</u>

กำหนดให้จำนวนโหนดในเซ็ตของโหนดปกติคุณภาพสูงมีจำนวนทั้งหมด n โหนด แต่ละโหนดต้องทำการพิจารณาลิงก์ทั้งหมดสูงสุด k ลิงก์ และในหนึ่งครั้งที่พิจารณาลิงก์ ต้องค้นหาในกลุ่มโหนดที่เป็นสแปมทั้งหมดจำนวน m โหนด ว่าต้องเปลี่ยนป้ายชื่อหรือไม่ ดังนั้นความซับซ้อนของอัลกอริธึมนี้สามารถเขียนให้อยู่ในรูป O(kmn)

3.7.4 **สรุปอัลกอริธึมรวม**

เมื่อเราประกอบอัลกอริธึมย่อยทั้ง 3 ส่วนเข้าด้วยกันเพื่อค้นหาโหนดที่เป็นสแปมแล้ว เราอาจจะสรุประบบทั้งหมดได้ดังนี้

ข้อมูลนำเข้า (input)

1. ข้อมูลชุดสอน (training data) ซึ่งประกอบไปด้วย
    a. ข้อมูลโหนดที่เป็นสแปมจำนวนหนึ่ง
    b. ข้อมูลโหนดที่เป็นโหนดปกติซึ่งมีคุณภาพสูง
2. ข้อมูลเว็บกราฟทั้งหมด ทีจะนำมาค้นหาสแปม อยู่ในรูปโฮสกราฟ

ผลลัพธ์ (result) : ระบุว่าโหนดใดบ้างเป็นสแปม

<u>วิเคราะห์ความซับซ้อน</u>

เนื่องจากเดิมทีนั้นในแต่ละส่วนของระบบถูกวิเคราะห์ความซับซ้อนด้วยตัวแปรที่แตกต่างกัน ดังนั้นเมื่อนำระบบมาประกอบรวมกัน จึงต้องนำแต่ละตัวแปรเข้ามาพิจารณาขณะประกอบโดยทำให้อยู่ในรูปเดียวกัน เพื่อให้เปรียบเทียบกันได้ชัดเจน โดยขั้นตอนดังต่อไปนี้

กำหนดให้

- ข้อมูลเว็บกราฟทั้งหมดมีจำนวนโหนด N โหนด มีจำนวนลิงก์ L ลิงก์
- จำนวนโหนดในข้อมูลสอนที่เป็นโหนดปกติคุณภาพสูง มีจำนวน $\frac{N}{r_n}$ โหนด
- จำนวนโหนดในข้อมูลสอนที่เป็นโหนดสแปม มีจำนวน $\frac{N}{r_s}$

จากข้อกำหนดดังกล่าวจะเห็นว่า r_n และ r_s คืออัตราส่วนระหว่างโหนดที่บรรจุในข้อมูลสอน ในส่วนโหนดปกติคุณภาพสูง และส่วนที่เป็นโหนดสแปม ต่อจำนวนโหนดทั้งหมด ตามลำดับ

เมื่อเราวิเคราะห์ความซับซ้อนด้วยการนับจำนวนครั้งของการอ่านข้อมูลที่เกิดขึ้น ในกรณีที่แย่ที่สุด จะได้จำนวนครั้งของการเปรียบเทียบดังนี้

1. ในขั้นตอนการตรวจหาบูสต์โหนด จากเดิมที่ความซับซ้อนของอัลกอริธึมเป็น O(mn) เมื่อ m คือจำนวนลิงก์และ n คือจำนวนโหนดสแปมในชุดข้อมูลสอน เมื่อแทนจำนวนลิงก์ด้วย L และจำนวนโหนดสแปมที่ใช้ในชุดข้อมูลสอนด้วย $\frac{N}{r_s}$ จะได้ความซับซ้อนในรูปใหม่เป็น $\frac{LN}{r_s}$

2. ในขั้นตอนการค้นหาโหนดสแปม จากเดิมที่มีความซับซ้อน O(n) เมื่อ n เป็นจำนวนลิงก์ในกลุ่มโหนดที่เป็นบูสต์โหนด ในกรณีแย่สุด (worst-case scenario) คือทุกๆ โหนดในเว็บกราฟเป็นบูสต์โหนดทั้งหมด และทุกๆ ลิงก์ในเว็บกราฟเป็นลิงก์ของบูสต์โหนด ดังนั้นจึงจำเป็นต้องอ่านข้อมูลเพื่อบันทึกทั้งหมด L ครั้ง

3. ในขั้นตอนการกรองโหนดปกติ จากเดิมที่มีความซับซ้อนเป็น O(kmn) เมื่อ k เป็นจำนวนลิงก์สูงสุดที่พิจารณาในแต่ละโหนดปกติ n คือจำนวนโหนดปกติ และ m คือจำนวนโหนดที่ตรวจสอบพบว่าเป็นสแปม ในกรณีแย่สุด คือเดิมทีทุกโหนดในเว็บกราฟเป็นโหนดที่ตรวจสอบพบว่าเป็นสแปมสแปม เมื่อแทนค่าจะได้รูปใหม่เป็น $k \times \frac{N}{r_n} x N$ หรือ $\frac{kN^2}{r_n}$

4. เมื่อรวมทั้งสามขั้นตอนเข้าด้วยกัน และกำหนดให้ r_s , r_n และ k เป็นค่าคงที่ จะได้ว่าความซับซ้อนของอัลกอริธึมรวมคือ

$$O(N^2 + LN + L)$$

เมื่อ N คือ จำนวนโหนดทั้งหมดในเว็บกราฟ และ L คือจำนวนลิงก์ทั้งหมดในเว็บกราฟ

3.8 การวัดและการทดสอบประสิทธิภาพ

การวัดและทดสอบประสิทธิภาพการทำงานทำโดยการเปรียบเทียบผลลัพธ์ที่ได้จากอัลกอริธึมกับชุดข้อมูลทดสอบที่ติดฉลากเรียบร้อยแล้วว่าเป็นโฮสสแปมหรือโฮสปกติ โดยกำหนดค่าที่ใช้ในการวัดประสิทธิภาพและวิธีการวัดประสิทธิภาพที่แตกต่างกัน

3.8.1 ค่าที่ใช้วัดประสิทธิภาพของระบบโดบรวม

ค่าความแม่นยำ (precision) เป็นค่าที่บ่งบอกความแม่นในการตรวจจับเว็บสแปมของระบบ ซึ่งค่านี้เป็นตัวชี้วัดและสะท้อนถึงโอกาสและความน่าจะเป็นที่เว็บสแปมในผลลัพธ์ของระบบจะเป็นเว็บสแปมจริง มิใช่เป็นเว็บปกติที่ระบบชี้ผิดพลาด นิยามโดย

$$Precision = \frac{The\ number\ of\ spam\ nodes\ found\ in\ result\ set}{The\ number\ of\ all\ nodes\ in\ result\ set}$$

ค่าเรียกคืน (recall) เป็นค่าที่บ่งบอกความละเอียดในการตรวจจับเว็บสแปมของระบบ ค่านี้เป็นตัวชี้วัดว่าระบบมีความละเอียดในการตรวจจับมากแค่ไหน ค่านี้จะเป็นอัตราส่วนผกผันกับโอกาสที่เว็บสแปมจะหลุดรอดไปจากการตรวจจับของระบบ นิยามโดย

$$Recall = \frac{The\ number\ of\ spam\ nodes\ found\ in\ result\ set}{The\ number\ of\ all\ spam\ nodes\ in\ the\ webgraph\ data}$$

ค่าเรียกคืนและค่าความแม่นยำนี้ชี้วัดประสิทธิภาพของระบบในมุมที่แตกต่างกัน กล่าวคือ ระบบที่มีค่าเรียกคือสูงสามารถรับประกันได้ว่าเว็บสแปมที่จะหลุดรอดจากระบบนี้ได้มีจำนวนน้อยมาก แต่ไม่อาจรับประกันได้ว่าเว็บที่ระบบนี้กล่าวหาว่าเป็นเว็บสแปมจะมีโอกาสเป็นเว็บสแปมจริงมากน้อยแค่ไหนอย่างไร ส่วนระบบที่มีค่าความแม่นยำสูงสามารถรับรองได้ว่ามีโอกาสน้อยมากที่ระบบจะระบุให้เว็บธรรมดาเป็นเว็บสแปม แต่ในทางตรงกันข้ามไม่สามารถรับรองได้ว่ามีเว็บสแปมที่หลุดรอดจากการตรวจจับมากน้อยแค่ไหนอย่างไร ซึ่งโดยทั่วไปแล้วหากกำหนดให้ระบบมีค่าเรียกคืนมากขึ้นจะทำให้ความแม่นยำต่ำลง

3.8.2 วิธีการวัดประสิทธิภาพการทำงานของระบบ

สำหรับการตรวจวัดประสิทธิภาพของระบบนั้นจะมองในมุมมองที่แตกต่างกันหลายมุมมองดังนี้

1. **การวิเคราะห์ความไวต่อจำนวนข้อมูลสอน**

เนื่องจากว่าระบบตรวจจับสแปมในงานวิจัยนี้จำเป็นต้องมีข้อมูลสอน ดังนั้นคำถามที่สำคัญอย่างหนึ่งคือตัวประสิทธิภาพของระบบขึ้นอยู่กับข้อมูลสอนมากขนาดไหน ระบบการตรวจจับสแปมแบบใช้ข้อมูลสอนที่ดีจะทำงานได้ดีหรือไม่หากข้อมูลสอนมีเปลี่ยนแปลงไป แล้วทำได้ผลดีอย่างไรเมื่อมีการเปลี่ยนแปลง ซึ่งการทดสอบนี้เราทำโดยทำการสุ่มตัดข้อมูลที่ใช้ในการ

สอนให้มีจำนวนน้อยลง แล้วนำมาเปรียบเทียบว่าเมื่อตัดข้อมูลออกไปให้น้อยลงร้อยละ 10 ต่อหนึ่งการทดลอง จนหมดข้อมูลเริ่มต้น แล้วจึงวิเคราะห์ความเปลี่ยนแปลงในเชิงประสิทธิภาพของอัลกอริธึม

2. **การวัดความแม่นยำในการตรวจจับสแปมโดยไม่มีการคัดกรองโหนดปกติ**

เราจะทำการเปรียบเทียบระหว่างการใช้ระบบโดยไม่มีการคัดกรองโหนดปกติกับระบบทั้งหมดโดยรวม เพื่อเสนอถึงเหตุและผลรวมไปถึงวิเคราะห์ความคุ้มค่าในการเพิ่มระบบการคัดกรองโดยการตรวจจับโหนดปกติ ซึ่งสามารถวัดโดยการเปรียบเทียบผลลัพธ์ของค่าเรียกคืนและค่าความแม่นยำของระบบ ก่อนมีการกรองโหนดปกติ

3. **การวัดความแม่นยำในการตรวจจับโหนดปกติอย่างแน่นอน**

เนื่องจากระบบนี้มีการใช้การตรวจจับโหนดปกติอย่างแน่นอน ซึ่งเราจะวัดว่าความแม่นยำของระบบและความสามารถในการตรวจจับโหนดปกติของระบบนั้นเป็นอย่างไรบ้าง และสามารถช่วยเพิ่มความแม่นยำแก่ระบบการตรวจจับสแปมได้มากน้อยอย่างไร

บทที่ 4

การทดลองและวิเคราะห์ผล

ในส่วนของผลการทดลองนี้จะอธิบายชุดข้อมูลที่ใช้ในการทดลอง ค่าพารามิเตอร์ต่างๆ ในการทดลอง และวิเคราะห์สาเหตุที่มาที่ไปของผลลัพธ์ในการทดลอง และเปรียบเทียบประสิทธิภาพกับอัลกอริธึมอื่นที่ใช้ข้อมูลชุดเดียวกันในแง่มุมต่างๆ

4.1 ชุดข้อมูลที่ใช้ในการทดสอบ

สำหรับชุดข้อมูลที่ใช้นั้นใช้ข้อมูลเว็บกราฟมาตรฐานซึ่งเก็บโดยฝ่ายวิจัยยาฮู [17] ข้อมูลดังกล่าวเป็นข้อมูลที่ได้จากการเก็บเพจในโดเมนของประเทศอังกฤษ ในปี 2006 ประกอบไปด้วยเพจ 77,714,046 เพจ 2,965,197,340 ไฮเปอร์ลิงก์ และ 11,402 โฮส ซึ่งใช้อาสาสมัครจำนวน 33 คน ในการแยกแยะประเภทว่าโฮสใดบ้างที่เป็นโฮสสแปม มีการกำหนดแนวทางในการจัดแยกประเภทโดยผู้เชี่ยวชาญ ซึ่งข้อมูลดังกล่าวได้แบ่งออกเป็นข้อมูลชุดสอนและข้อมูลทดสอบ เพื่อใช้ในการเปรียบเทียบประสิทธิภาพกับระบบการตรวจจับเว็บสแปมที่ใช้วิธีการการเรียนรู้ของเครื่องได้ ในข้อมูลชุดทดสอบประกอบด้วยโฮสสแปมทั้งหมด 1250 โหนด จากโฮสทั้งหมด 1851 โฮส

4.2 สภาพแวดล้อมในการทดสอบ

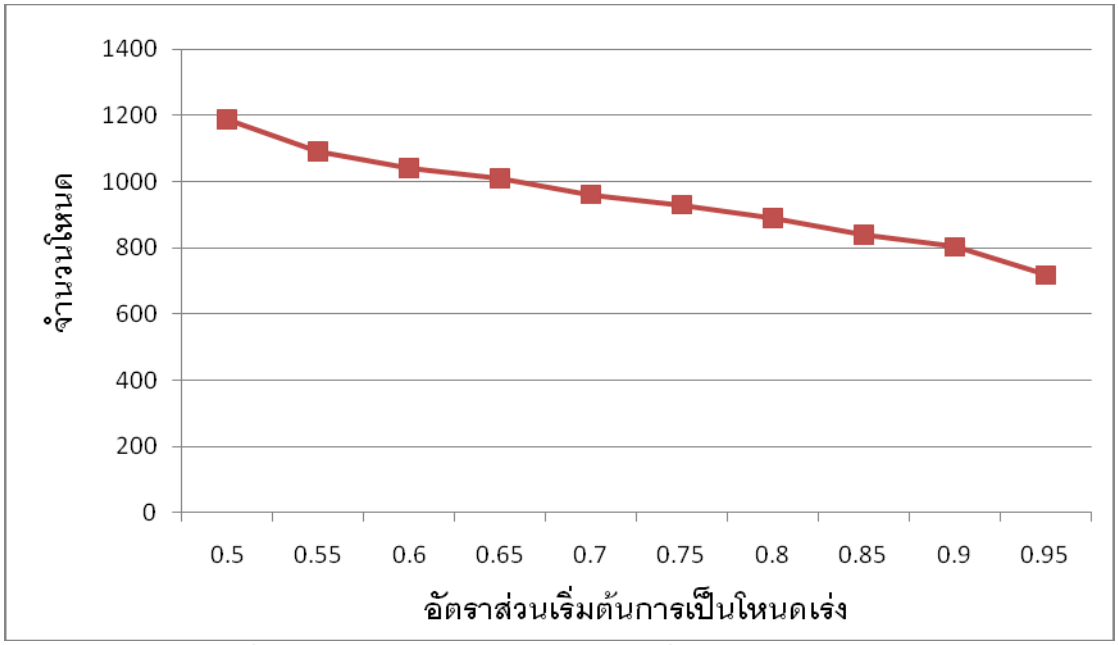
ระบบการทดสอบทำบนโปรแกรมไมโครซอฟต์วิชวลสตูดิโอ (Microsoft Visual Studio 2010) โดยใช้ภาษาซีชาร์ป (C#) ในการเขียน โดยเครื่องคอมพิวเจอร์ที่ใช้ในการทดสอบคือ คอมพิวเตอร์โน้ตบุ๊กเอเซอร์ แอสไพร์ (Acer Aspire) รุ่น 4520G โดยมีสเปกของเครื่องคือ

- ระบบปฏิบัติการวินโดวส์ เอ็กซ์พี เซอร์วิซแพค 3 (Windows XP Service Pack 3)
- หน่วยประมวลผลเอเอ็มดีทูรอน 64 ความเร็ว 1.9 GHz พร้อมแอลทูแคช 512 KB 2 ตัว
- หน่วยความจำดีดีอาร์-2 (DDR2) ขนาดทั้งหมด 2 GB

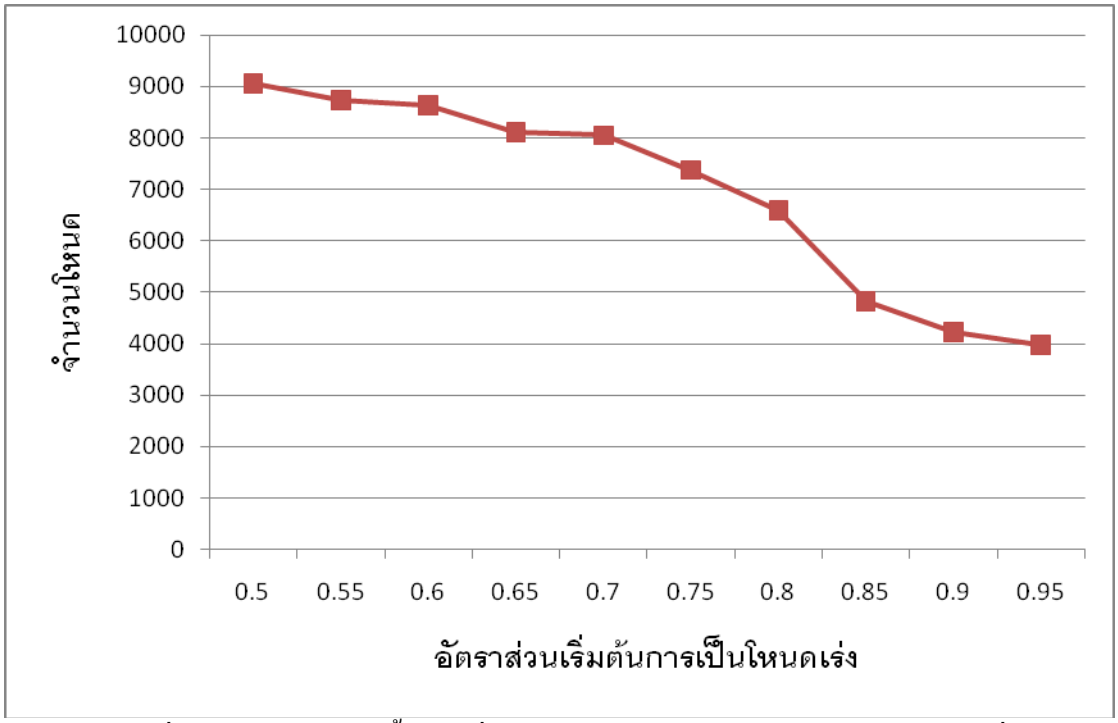
4.3 ความละเอียดในการตรวจจับบูสต์โหนด

การกำหนดค่าอัตราส่วนเริ่มต้นการเป็นโหนดเร่งนั้นมีผลต่อประสิทธิภาพการทำงานของระบบเป็นอย่างมาก หากกำหนดค่าไว้สูงเกินไปจะทำให้บูสต์โหนดที่ได้จากข้อมูลสอนไม่สามารถ

นำไปขยายผลตรวจจับเว็บสแปมได้ครบถ้วน ดังนั้นค่าที่เหมาะสมนั้นเราสามารถดูได้จากจำนวนบูสต์โหนดที่ได้จากอัตราส่วนเริ่มต้นการเป็นโหนดเร่งที่แตกต่างกัน และจำนวนโหนดที่สามารถขยายผลได้ในอัตราส่วนเริ่มต้นการเป็นโหนดเร่งต่างๆ ซึ่งได้ผลการทดลองดังภาพที่ 4.1 และ 4.2



ภาพที่ 4.1 จำนวนโหนดเร่งบนอัตราส่วนเริ่มต้นการเป็นโหนดเร่งแต่ละค่า



ภาพที่ 4.2 จำนวนโหนดทั้งหมดที่สามารถเข้าถึงได้ผ่านโหนดเร่งบนอัตราส่วนเริ่มต้นการเป็นโหนดเร่งแต่ละค่า

จากภาพที่ 4.1 และ 4.2 จะเห็นว่าถึงแม้ว่าจำนวนโหนดเร่งที่ลดลงจากการปรับอัตราส่วนเริ่มต้นการเป็นโหนดเร่งมีไม่มากนัก แต่จำนวนโหนดที่สามารถเข้าถึงได้นั้นกลับลดลงมาก ซึ่งการลดลงของจำนวนโหนดที่เข้าถึงได้นั้นแปลได้ว่าจำนวนโหนดที่ผ่านการตรวจสอบมีน้อยลง ในหมู่โหนดที่เข้าถึงไม่ได้นั้นมีทั้งโหนดที่เป็นสแปมและโหนดปกติ การปรับให้ค่าอัตราส่วนเริ่มต้นสูงขึ้นจะมีข้อดีคือทำให้โหนดปกติที่ไม่เป็นสแปมถูกกรองให้เข้าถึงไม่ได้ตั้งแต่ต้น แต่ในทางกลับกันก็จะทำให้โหนดที่เป็นสแปมบางตัวนั้นถูกกรองออกให้เข้าถึงไม่ได้เช่นกัน หรืออีกนัยหนึ่งคือไม่สามารถตรวจสอบความเป็นสแปมได้โดยระบบ

แต่เนื่องจากในระบบของเรานั้นมีการกรองโหนดปกติในภายหลังอยู่แล้ว ดังนั้นการปรับอัตราส่วนเริ่มต้นการเป็นโหนดเร่งให้สูงเพื่อช่วยกรองโหนดปกติให้เข้าถึงไม่ได้ตั้งแต่แรก จึงไม่มีความจำเป็นมากนัก ซึ่งในการทดลองถัดมาจะมีผลชัดเจนว่าการกรองโหนดปกติโดยการปรับอัตราส่วนเริ่มต้นการเป็นโหนดเร่ง ด้อยประสิทธิภาพกว่าการคัดกรองโดยการค้นหาโหนดปกติอย่างแน่นอนเป็นอย่างมาก

4.4 ประสิทธิภาพของระบบการตรวจจับสแปมแบบไม่คัดกรองโหนดปกติ

ระบบการตรวจจับสแปมแรกเริ่ม จะยังไม่มีการนำระบบการคัดกรองโหนดปกติเข้ามาใช้ ซึ่งเราสามารถวัดประสิทธิภาพก่อนเพิ่มระบบคัดกรองโหนดปกติ ได้ดังตารางที่ 4.1

Thereshold	Spam	Normal	Recall	Precision
0.5	1165	292	93.20%	79.96%
0.6	1097	249	87.76%	81.50%
0.7	1011	193	80.88%	83.97%
0.8	792	133	63.36%	85.62%
0.9	595	117	47.60%	83.57%

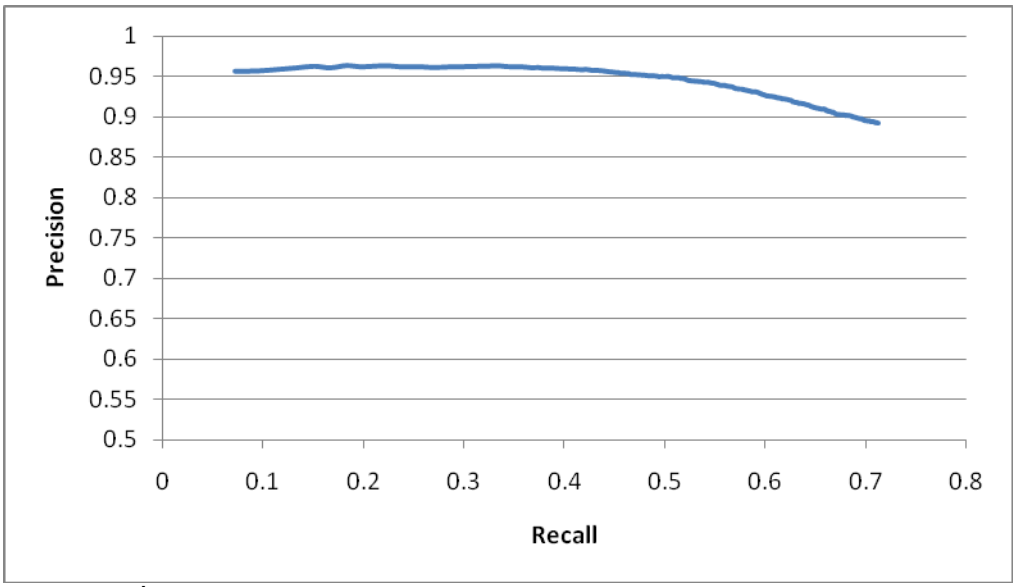
ตารางที่ 4.1 ผลการตรวจจับเว็บสแปมโดยไม่คัดกรองโหนดปกติ

จากตารางที่ 4.1 จะเห็นว่าอัลกอริธึมการตรวจจับสแปมนั้นทำงานได้ผลดีบนอัตราส่วนเริ่มต้นการเป็นโหนดเร่งอยู่สูงว่า 0.7 ทั้งในแง่ของค่าความแม่นยำและค่าเรียกคืน (สูงกว่า 75%) แต่ทั้งนี้แล้วหากเราต้องการเพิ่มความแม่นยำโดยการลดอัตราส่วนเริ่มต้นการเป็นโหนดเร่ง จะพบว่าค่าความแม่นยำไม่เปลี่ยนแปลงมากนัก แต่ค่าเรียกคืนนั้นกลับลดลงเป็นอย่างมาก ซึ่งทำให้

การเพิ่มความละเอียดถูกต้องโดยการลดค่าอัตราส่วนการเป็นโหนดเร่งนั้นเป็นทางเลือกที่ไม่เหมาะสม

4.5 การทดสอบประสิทธิภาพของระบบการคัดเลือกโหนดปกติ

สำหรับระบบการคัดเลือกโหนดปกตินั้นจะทดสอบโดยการพิจารณาว่าระบบการคัดโหนดปกติที่ได้จากระบบนั้นมีความแม่นยำมากแค่ไหนเมื่อเปลี่ยนค่าลิงก์สูงสุด k ตัวเป็นค่าต่างๆ ซึ่งผลลัพธ์ที่ได้เป็นดังกราฟในภาพที่ 4.3 และตารางที่ 4.2



ภาพที่ 4.3 กราฟแสดงความสัมพันธ์ระหว่างค่าเรียกคืนและค่าความแม่นยำของระบบตรวจจับโหนดปกติ

K	โหนดที่พบ	โหนดปกติที่ค้นพบ
1	615	588
2	804	769
3	990	949
4	1136	1091
5	1275	1227
6	1414	1358
7	1547	1490
8	1666	1602
9	1787	1720
10	1890	1820

K	โหนดที่พบ	โหนดปกติที่ค้นพบ
11	2004	1927
12	2101	2020
13	2200	2115
14	2290	2200
15	2376	2284
16	2453	2359
17	2541	2444
18	2614	2515
19	2679	2578
20	2754	2651
∞	8459	7287

ตารางที่ 4.2 จำนวนโหนดที่ได้และจำนวนโหนดปกติที่ค้นพบผ่านอัลกอริธึมการตรวจค้นโหนดปกติ บนค่า k ต่างๆ

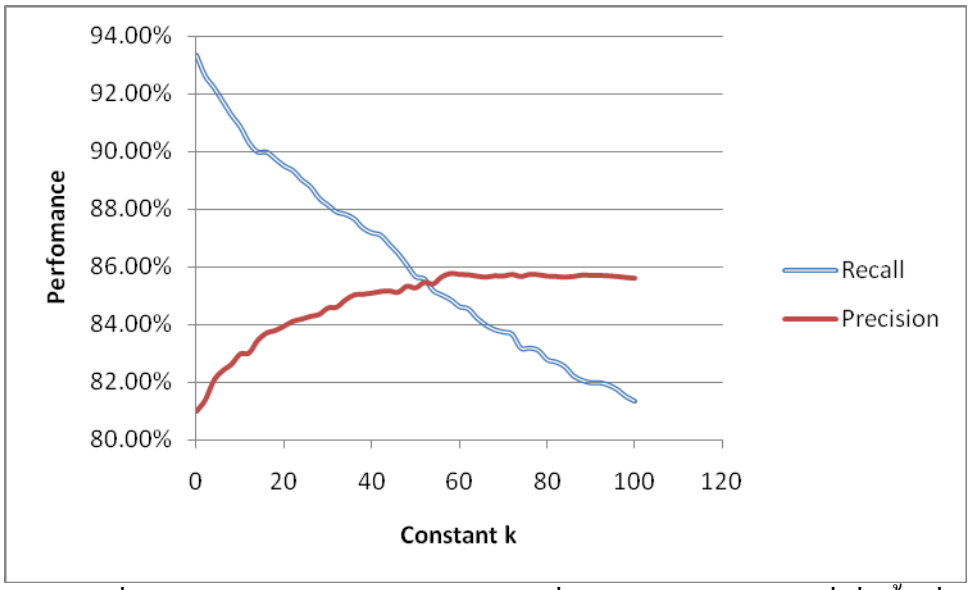
จากผลการทดลองพบว่าการค้นหาโหนดปกติสามารถหาได้ด้วยความแม่นยำสูงสุดถึง 96.30% สูงสุดระดับค่าเรียกคืน 33.33% แต่ทั้งนี้หากต้องการดึงโหนดปกติให้มากที่สุดก็สามารถเรียกโหนดได้มากถึง 89.71% ในระดับค่าความแม่นยำ 86.14% เช่นกัน แต่ทั้งนี้ในระบบการตรวจจับสแปมนั้น การมีข้อผิดพลาดประเภท 1 นั้นจะทำให้โหนดที่เป็นสแปมหลุดจากการตรวจสอบ ซึ่งส่งผลให้ระบบการตรวจจับสแปมมีประสิทธิภาพลดลง แต่การมีข้อผิดพลาดประเภท 2 นั้นเมื่อใช้ร่วมกับระบบตรวจจับสแปมแล้วจะไม่ส่งผลให้ระบบการตรวจจับสแปมมีประสิทธิภาพลดลงแต่อย่างใด จึงยอมรับข้อผิดพลาดประเภทที่ 2 ได้มากกว่า กล่าวคือ ระบบต้องให้ความสำคัญกับความแม่นยำในผลลัพธ์ มากกว่าความสามารถในการตรวจจับโหนดปกติได้เป็นจำนวนมาก

4.6 การทดสอบประสิทธิภาพของระบบโดยรวม

การทดสอบประสิทธิภาพของระบบโดยรวมนั้นทำโดยการนำระบบทั้งหมดมาประกอบรวมกัน และทดสอบประสิทธิภาพในมุมต่างๆ ดังนี้

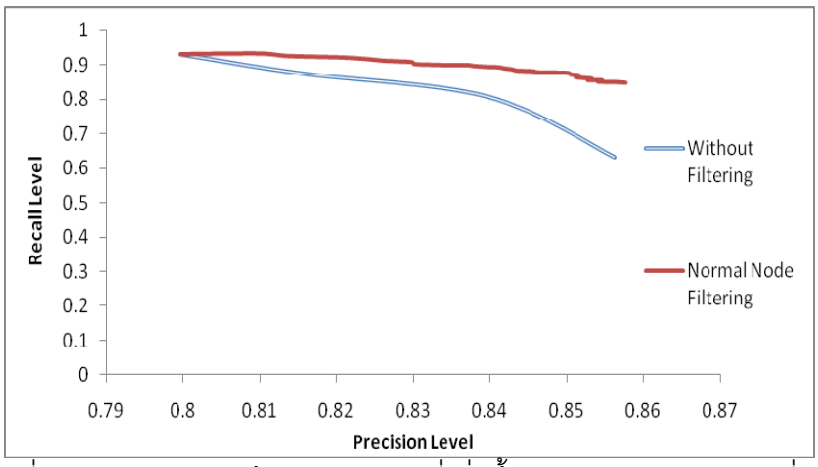
1. เปรียบเทียบประสิทธิภาพเมื่อเมื่อปรับค่าลิงก์สูงสุด k ตัวที่โหนดปกติยอมรับ ดังภาพที่ 4.4

2. เปรียบเทียบประสิทธิภาพที่เพิ่มขึ้นเมื่อใช้ระบบการตรวจคัดกรองโหนดปกติ เทียบกับการปรับค่าอัตราส่วนเริ่มต้นการเป็นโหนดเร่ง ดังภาพที่ 4.5



ภาพที่ 4.4 แผนภูมิแสดงถึงอัตราค่าเรียกคืนที่ลดลงและความแม่นยำที่เพิ่มขึ้นเมื่อเพิ่มค่าคงที่ k

จากภาพที่ 4.4 จะเห็นว่าเมื่อเราเพิ่มค่าคงที่จำนวนลิงก์สูงสุดที่พิจารณา k จะทำให้ค่าเรียกคืนลดลงและค่าความแม่นยำเพิ่มขึ้น เพราะเนื่องจากเราทำการตัดโหนดบางส่วนออกจากผลลัพธ์ของเว็บสแปมเดิม ซึ่งจะทำให้ค่าเรียกคืนลดลงจากการที่มีโอกาสตัดผิดพลาดไปตัดโหนดที่เป็นสแปมออกจากผลลัพธ์ และค่าความแม่นยำเพิ่มขึ้นจากการตัดโหนดปกติออกจากผลลัพธ์ ซึ่งจะเห็นว่าค่าความแม่นยำนั้นจะมีการอัตราการเพิ่มขึ้นอย่างเร็วในช่วงแรก แล้วจึงมีอัตราการเพิ่มขึ้นค่อยๆ เริ่มลดลงมาเมื่อ k มีค่าเป็น 20 จนเริ่มหยุดดนิ่งเมื่อค่า k เป็น 60



ภาพที่ 4.5 แผนภูมิแสดงถึงประสิทธิภาพที่เพิ่มขึ้นในแง่ของจำนวนโหนดที่ตรวจพบ เมื่อกำหนดระดับความแม่นยำให้สูงขึ้น

จากผลการทดลองตามภาพที่ 4.5 พบว่า ในระดับความแม่นยำช่วง 80% นั้น ไม่มีความแตกต่างระหว่างการกรองโหนดปกติช่วยกับระบบแบบไม่กรองโหนดปกติ แต่เมื่อต้องการเพิ่มความแม่นยำมากขึ้น จะเห็นได้ชัดว่าการเพิ่มความแม่นยำโดยการเพิ่มอัตราส่วนเริ่มต้นการเป็นโหนดเร่งนั้นส่งผลให้โหนดสแปมที่ตรวจจับได้ลดลงมากกว่าการกรองโหนดปกติอย่างเห็นได้ชัด ดังนั้นการปรับระดับความแม่นยำของระบบควรใช้การกรองโหนดปกติมากกว่าการปรับค่าอัตราส่วนเริ่มต้นการเป็นโหนดเร่ง

4.7 ความละเอียดอ่อนของความแม่นยำต่อจำนวนข้อมูลสอน

ในการวัดความละเอียดต่อข้อมูลสอน จะวัดใน 2 แง่มุม คือในแง่ของความละเอียดอ่อนต่อจำนวนข้อมูลสแปมที่ใช้ในการสอน และในแง่ของความละเอียดต่อจำนวนข้อมูลโหนดปกติที่ใช้ในการสอน โดยกำหนดให้ค่าคงที่ลิงก์สูงสุด k เป็นค่าคงตัวเพื่อเป็นตัวแปรควบคุมไม่ให้มีผลต่อความละเอียดอ่อน

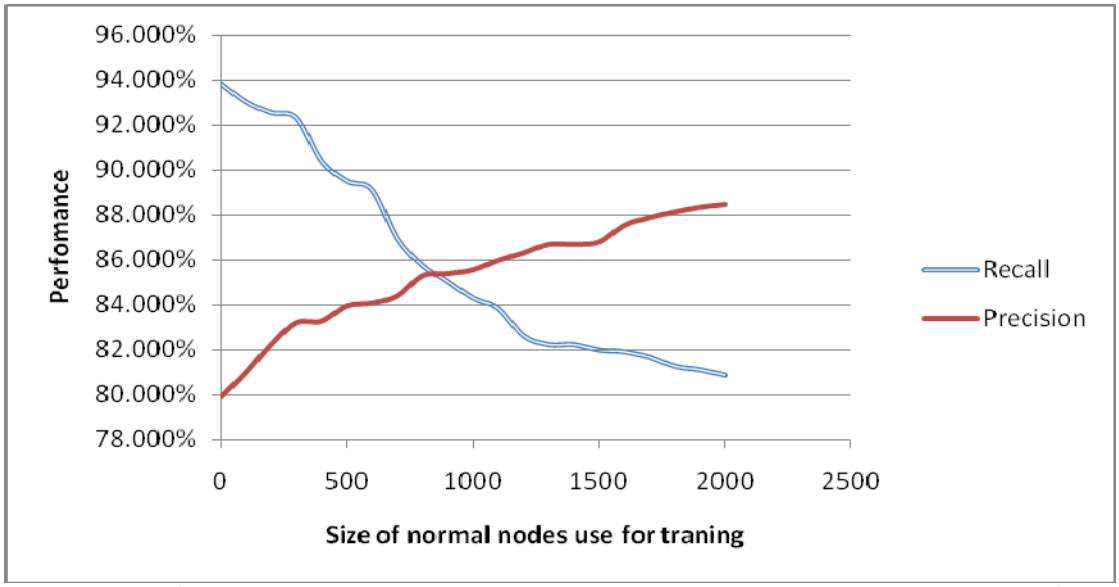
ในการวัดความละเอียดอ่อนต่อจำนวนข้อมูลสแปม เพื่อตัดปัจจัยความแม่นยำจากการเพิ่มโหนดปกติ เราจะทำการทดสอบโดยใช้เพียงอัลกอริธึมการตรวจจับสแปมโดยไม่คัดกรองโหนดปกติ โดยได้ทำการเปลี่ยนลดขนาดข้อมูลชุดสอนให้ใช้เพียง 90%, 80%, 70% ของชุดข้อมูลสอนทั้งหมด โดยตัดลดลงไปเรื่อยๆ โดยการสุ่มตัดข้อมูลสอนบางส่วนออก แล้ววัดประสิทธิภาพที่ได้ ซึ่งผลการทดลองเป็นไปดังภาพที่ 4.6

ภาพที่ 4.6 ประสิทธิภาพการทำงานเมื่อเปลี่ยนจำนวนชุดข้อมูลสอน

ผลการทดลองชี้ให้เห็นว่า หากข้อมูลชุดสอนมีจำนวนน้อยไป จำนวนโหนดสแปมที่สามารถค้นพบจะมีจำนวนลดลงมากโดยสังเกตจากค่าเรียกคืนที่ลดลง แต่เมื่อข้อมูลชุดสอนมี

จำนวนมากพอในระดับหนึ่งจนสามารถเชื่อมค้นหาได้ทั้งกราฟแล้ว ประสิทธิภาพในการตรวจจับสแปมจะไม่ขึ้นอยู่กับชุดข้อมูลสอนอีกต่อไป ทั้งในแง่ค่าเรียกคืนและค่าความแม่นยำจะไม่มีความเปลี่ยนแปลงมากนัก

ถัดมาเราจะตรวจสอบความละเอียดอ่อนต่อชุดข้อมูลสอนที่ใช้ในการตรวจจับโหนดปกติและผลกระทบต่อการตรวจจับสแปม ซึ่งเราได้ทำการกำหนดค่าคงที่อัตราส่วนเริ่มต้นการเป็นโหนดเร่งไว้ที่ 0.5 ใช้ค่าจำนวนลิงก์พิจารณาสูงสุด 10 ลิงก์ แล้วทดลองเปลี่ยนจำนวนข้อมูลสอนที่เป็นโหนดปกติ เพื่อดูผลกระทบต่อประสิทธิภาพรวมของระบบ ซึ่งผลการทดลองแสดงได้ดังรูป 4.7



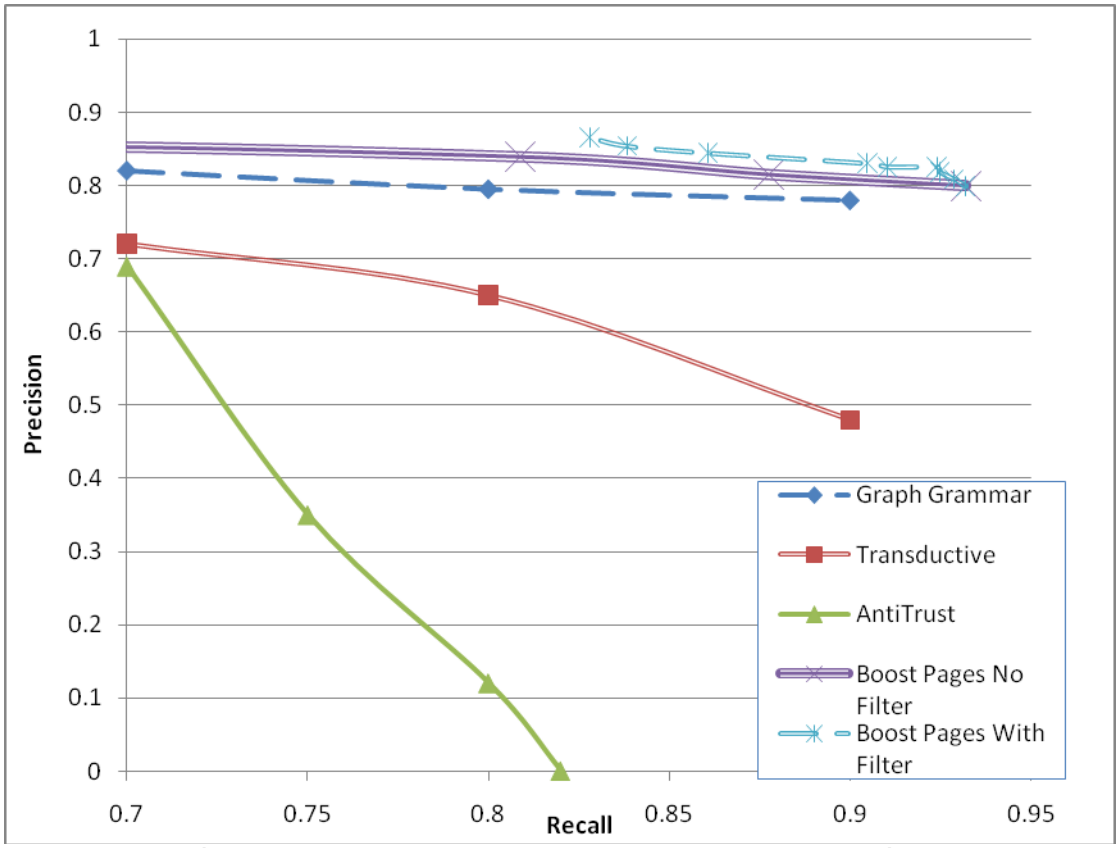
ภาพที่ 4.7 แผนภูมิแสดงความสัมพันธ์ระหว่างประสิทธิภาพและจำนวนโหนดปกติที่ใช้สอน

จากผลการทดลองจะเห็นว่าจำนวนโหนดปกติที่ใช้สอนนั้นแปรผันตรงกับค่าความแม่นยำและแปรผกผันกับค่าเรียกคืน ทั้งนี้เพราะหากมีการใช้โหนดปกติเป็นจำนวนมากก็จะเพิ่มโอกาสที่โหนดที่เป็นสแปมจะถูกชี้โดยโหนดปกติในกลุ่มสอน จึงทำให้เพิ่มโอกาสผิดพลาดที่จะหาโหนดสแปมไม่เจอ แต่กลับกันก็เพิ่มโอกาสที่ที่โหนดปกติจะถูกชี้โดยโหนดปกติในกลุ่มสอน ค่าความแม่นยำจึงเพิ่มขึ้นเช่นกัน

4.8 ผลการเปรียบเทียบประสิทธิภาพกับอัลกอริธึมอื่น

สำหรับการเปรียบเทียบประสิทธิภาพกับอัลกอริธึมอื่นที่ใช้ชุดข้อมูลเดียวกัน จะทำการทดสอบในแง่ที่ว่าตรวจสอบความแม่นยำในระดับค่าเรียกคืนที่แตกต่างกัน ซึ่งวัดผลโดยกำหนดพารามิเตอร์อัตราส่วนเริ่มต้นการเป็นโหนดเร่งที่ 0.5 และใช้จำนวนโหนดเร่งคุณภาพสูงทั้งหมด 100 ตัว กำหนดค่าลิงก์สูงสุด k เป็น 20 ซึ่งเป็นค่าที่เริ่มส่งผลให้อัตราการเพิ่มความแม่นยำลด

น้อยลงตามภาพที่ 4.4 แล้วจึงนำมาพล็อตแผนภูมิความสัมพันธ์ระหว่างค่าเรียกคืนและค่าความแม่นยำในระดับต่างๆ ดังภาพที่ 4.8



ภาพที่ 4.8 ผลการเปรียบเทียบความแม่นยำบนระดับความเรียกคืนที่แตกต่างกันของอัลกอริธึมการตรวจจับสแปมต่างๆ

จากผลการทดลองจะเห็นว่าในระดับค่าเรียกคืนที่สูง วิธีการในงานวิจัยจะให้ผลลัพธ์ค่าความแม่นยำที่สูงกว่า

4.9 วิเคราะห์ผลการทดลอง

จากผลการทดลองพบว่าเราสามารถใช้การค้นหาบูสต์โหนดในการตรวจจับสแปมได้ ซึ่งประสิทธิภาพนั้นขึ้นอยู่กับความละเอียดของการคัดสรรบูสต์โหนดเป็นอย่างมาก หากกำหนดค่าอัตราส่วนเริ่มต้นการเป็นโหนดเร่งไว้สูง จะทำให้โหนดเร่งในชุดข้อมูลเราไม่สามารถขยายผลไปทั่วถึงเว็บกราฟ ทำให้มีเว็บสแปมจำนวนมากในเว็บกราฟที่ไม่ได้รับการตรวจจับและหลุดพ้นไป แต่หากกำหนดไว้ต่ำเกินไป ก็จะส่งผลให้มีเว็บธรรมดาได้รับผลกระทบจากการตรวจจับเว็บสแปมมากขึ้น

เว็บสแปมจำนวนมากมิได้ได้รับคะแนนเร่งจากบูสต์โหนดเพียงอย่างเดียว ซึ่งทำให้การตรวจจับสแปมโดยกำหนดลงไปทันทีว่าเว็บที่ได้รับคะแนนจากบูสต์โหนดเป็นสแปม แล้วจึงคัดกรองเว็บที่ดีออกไปทีหลัง ทำได้ง่ายกว่าและลดประสิทธิภาพในเชิงความละเอียดในการตรวจจับน้อยกว่าการพยายามเลือกบูสต์โหนดที่ชี้ไปหาสแปมเพียงอย่างเดียว

ในการเลือกข้อมูลสอนที่จะทำให้การตรวจบูสต์เพจสามารถตรวจจับสแปมได้ทั่วถึงเว็บกราฟ จำเป็นต้องใช้ข้อมูลสอนจำนวนหนึ่ง ซึ่งทำให้ในตอนแรกหากข้อมูลสอนไม่เพียงพอระบบจะตรวจจับสแปมได้ไม่ดีนัก แต่เมื่อข้อมูลสอนมีมากพอแล้วประสิทธิภาพของระบบจะไม่ขึ้นอยู่กับขนาดของข้อมูลสอนอีกต่อไป

ในการเปรียบเทียบกับงานวิจัยที่เกี่ยวข้อง จะพบว่าในระดับค่าเรียกคืนประมาณ 80-90% เราสามารถหาผลลัพธ์ได้มีความแม่นยำสูงกว่าอัลกอริธึมเก่าทั้ง 3 ตัว แต่วิธีของเราไม่สามารถปรับให้มีค่าเรียกคืนได้สูงขึ้นไปกว่านี้ได้อีก และในระดับค่าเรียกคืนที่ต่ำ ค่าความแม่นยำของระบบจะไม่เพิ่มขึ้นอย่างมีนัยสำคัญ

บทที่ 5

สรุปผลงานวิจัยและข้อเสนอแนะ

5.1 สรุปผลงานวิจัย

งานวิจัยนี้ได้นำเสนอแนวคิดใหม่ในการตรวจจับเว็บสแปมโดยการค้นหาเว็บบูสต์เพจแล้วจึงนำไปใช้หาเว็บสแปมจริง ซึ่งต่างจากแนวคิดเดิมที่จะเน้นการค้นหาลักษณะของเพจสแปมเป็นหลัก โดยวิธีที่ใช้ในการค้นหาเว็บบูสต์เพจทำได้โดยการใช้ข้อมูลสอนจำนวนหนึ่งเพื่อค้นหาบูสต์เพจของข้อมูลสอน และเนื่องจากลักษณะของเว็บสแปมที่มักจะรวมเป็นสังคม (community) ทำให้เว็บบูสต์เพจที่ได้จากชุดข้อมูลสอนสามารถนำไปใช้ต่อยอดค้นหาเว็บเพจที่เป็นสแปมได้อีกจำนวนมาก

นอกจากนี้เนื่องจากว่าในการค้นหาบูสต์เพจจากข้อมูลสอนพบปัญหาว่าหากใช้บูสต์เพจที่ละเอียดเพื่อหลีกเลี่ยงมิให้เว็บปกติถูกลงโทษ จะไม่สามารถต่อยอดค้นหาเว็บเพจที่เป็นสแปมได้ครบถ้วนมากนัก และส่งผลกระทบต่อจำนวนเว็บสแปมที่ค้นพบค่อนข้างสูง เพราะบูสต์เพจที่ละเอียดเหล่านี้มักจะชี้เข้าหาเพจสแปมเพียงเพจเดียว ไม่สามารถใช้ในการกระจายผลลัพธ์ไปสู่เพจสแปมอื่นได้ เพื่อแก้ปัญหานี้จึงได้เสนอว่าหากต้องการหลีกเลี่ยงการลงโทษเว็บปกติแล้ว ควรใช้บูสต์เพจในระดับความละเอียดไม่สูง ค้นหาผลลัพธ์เว็บสแปมออกมาก่อน แล้วจึงค่อยทำการกรองเว็บปกติออกจากผลลัพธ์อีกที ซึ่งการทำเช่นนี้จะทำให้เว็บเพจปกติที่ถูกลงโทษนั้นมีจำนวนน้อยลง โดยไม่ส่งผลกระทบต่อจำนวนเว็บสแปมที่ค้นพบมากเท่าการใช้บูสต์เพจที่ละเอียด

ระบบทั้งหมดจะถูกแบ่งออกเป็นสามส่วน คือระบบการคัดกรองบูสต์เพจ ระบบการตรวจจับเพจสแปมจากบูสต์เพจ และระบบการคัดกรองโหนดปกติ ซึ่งแยกออกจากกันอย่างชัดเจน ในแต่ละระบบมีความซับซ้อนในเชิงเวลาอยู่ในรูปโพลิโนเมียลไทม์โดยมีดีกรีไม่เกินหนึ่ง และเมื่อร่วมทั้งสามระบบเข้าด้วยกันแล้วจะมีความซับซ้อนเชิงเวลาอยู่ในรูปโพลิโนเมียลไทม์โดยมีดีกรีไม่เกินสอง

จากผลการทดลองพบว่า ระบบการตรวจจับเว็บสแปมโดยวิเคราะห์บูสต์เพจสามารถตรวจจับเว็บสแปมได้แม่นยำและทั่วถึงเมื่อกำหนดค่าพารามิเตอร์อัตราส่วนเริ่มต้นการเป็นบูสต์โหนดไว้ในระดับไม่สูงมากเกินไป แต่ถ้าต้องการเพิ่มความแม่นยำในการตรวจจับแล้ว การเพิ่มระดับอัตราส่วนเริ่มต้นการเป็นบูสต์โหนด จะทำให้ประสิทธิภาพของระบบในแง่จำนวนของเว็บสแปมที่ตรวจเจอ ลดลงสูงกว่าการเพิ่มความแม่นยำด้วยการคัดกรองโหนดปกติมาก เมื่อเทียบกับระบบการตรวจจับสแปมแบบอื่นแล้ว ระบบการตรวจจับเว็บสแปมโดยวิเคราะห์บูสต์เพจนั้นมีจุดเด่นกว่าระบบที่นำมาเปรียบเทียบในแง่ของระดับค่าเรียกคืนที่ 80-95% จะสามารถวิเคราะห์ผลลัพธ์ได้แม่นยำกว่า แต่ทั้งนี้เมื่อเทียบกับระบบอื่นแล้วจะเห็นว่าใน

ระบบการตรวจจับสแปมโดยวิเคราะห์บูสต์เพจนี้ไม่สามารถเพิ่มระดับความแม่นยำให้สูงเกิน 86% ได้ ซึ่งต่างจากระบบอื่นที่เมื่อใช้ค่าเรียกคืนที่น้อยลงความแม่นยำจะเพิ่มขึ้น

ดังนั้นระบบการตรวจจับเว็บสแปมโดยใช้บูสต์เพจ จึงสามารถใช้ช่วยเพิ่มคุณภาพของผลลัพธ์การค้นคืนจากระบบสืบค้นได้เป็นอย่างดี และในขั้นเพิ่มเติมอาจสามารถนำผลลัพธ์ที่ได้นำไปใช้ช่วยให้เข้าใจถึงปัจจัยที่ทำให้แต่ละเว็บเพจเป็นเว็บเร่งคะแนนมากขึ้น เราอาจพัฒนาระบบการให้คะแนนของระบบค้นคืนให้มีความยุติธรรมกว่าเดิมได้

5.2 **ข้อเสนอแนะ**

1. ในการพัฒนาระบบการทำงานอาจจะไม่จำเป็นต้องพัฒนาใหม่ทั้งระบบ แต่พัฒนาเพียงบางส่วนของระบบได้ เช่นหาระบบตรวจจับสแปมจากบูสต์เพจที่ดีกว่าเดิม

2. ในการค้นหาเว็บปกติอย่างแน่นอน อาจใช้ปัจจัยโครงสร้างทางเนื้อหาเข้าช่วยเหลือ รวมไปถึงอาจจะสำรวจจากความพึงพอใจของผู้ใช้และพฤติกรรมของผู้ใช้ได้ เพื่อให้เป็นตัวช่วยเหลือในการสร้างเซ็ตเว็บปกติที่ใช้เป็นข้อมูลป้อน

3. การเพิ่มเติมระบบการค้นหาบูสต์เพจ อาจทำได้เพิ่มเติมโดยการถ่วงน้ำหนักให้แก่ลิงก์ที่มีคุณสมบัติบางประการเด่นชัดเป็นพิเศษ โดยศึกษาจากลักษณะโครงสร้างของลิงก์ ว่าโครงสร้างแบบไหนเน้นการเร่งคะแนนอย่างไม่ยุติธรรมบ้าง

4. การพัฒนาระบบการตรวจจับสแปมจากบูสต์เพจ จะทำได้ดีขึ้นหากมีการจัดประเภทหมวดหมู่ของเพจที่เรากำลังตรวจจับ เพราะเพจบางประเภทมีโอกาสสูงกว่าปกติที่ถูกฝังลิงก์เพื่อใช้ในการเร่งคะแนนและได้รับลิงก์จากบูสต์เพจ เช่น เว็บบอร์ด โซเชียลเน็ตเวิร์ก เป็นต้น การแยกประเภทของเว็บเพจจะทำให้สามารถถ่วงคะแนนให้มีความยุติธรรมแก่เว็บเพจประเภทพิเศษได้ดียิ่งขึ้น

รายการอ้างอิง

[1] L. Becchetti, C. Castillo, D. Donato, S. Leonardi, and R. Baeza-Yates. “Link-based characterization and detection of Web Spam”. In Proceedings *of the 2nd International Workshop on Adversarial Information Retrieval on the Web*. (2006).

[2] D. Zhou, C.Burges and T.Tao. “Transductive Link Spam Detection”. Proceedings of *the 3rd international workshop on Adversarial Information Retrieval on the Web*. (2007).

[3] D. Gibson, R. Kumar, and A. Tomkins. “Discovering Large Dense Subgraphs in Massive Graphs*”. In the Proceedings of the 31st International conference on Very Large Data Bases*. (2005).

[4] P.T. Metaxas and J. DeStefano. “Web Spam, Propaganda and Trust”. In the *Proceedings of the 1st International Workshop on Adversarial Information Retrieval on the Web*. (2005).

[5] H. Saito, M. Toyoda, M. Kitsuregawa, and K. Aihara. “A Large-Scale Study of Link Spam Detection by Graph Algorithms”. *In the Proceedings of the 3rd International Workshop on Adversarial Information Retrieval on the Web*. (2007).

[6] Z. Gyöngyi, H. Garcia-Molina, and J. Pedersen. “Combating Web Spam with TrustRank”. *In the Proceedings of the 30th International Conference on Very Large Data Bases*. (2004).

[7] Z. Gyöngyi, H. Garcia-Molina, and J. Pedersen, “Link Spam Detection Based on Mass Estimation”. In Proceeding of *32nd International Conference on Very Large Data Bases*.

[8] V. Krishnan and R. Raj. “Web Spam Detection with Anti-Trust Rank”. In *the Proceedings of the 2nd International Workshop on Adversarial Information Retrieval on the Web*. (2006).

[9] L. Pages, S. Brin, R. Motwani, and T. Winograd. “The PageRank Citation Ranking: Bring Order to the Web”. Technical report, Stanford Digital Libraries. (1998).

[10] Y. Du, Y. Shi, and X. Zhao. “Using Spam Farm to Boost PageRank”. In the *Proceedings of the 3rd International Workshop on Adversarial Information Retrieval on the Web*. (2007).

[11] Chobtham, K.; Surarerks, A.; Rungsawang, “A. Formalization of Link Farm Structure Using Graph Grammar”. *Advanced Information Networking and Applications,* 2008

[12] B. Wu and K. Chellapilla. “Extracting Link Spam using Biased Random Walks from Spam Seed Sets”. In the *Proceedings of the 3rd International Workshop on Adversarial Information Retrieval on the Web*. (2007).

[13] Z. Gyongyi and H. Garcia-Molina. “Link Spam Alliance”. *In Proceedings of 31st VLDB Conference*, Trondheim, Norway. 2005

[14] W. Wongsarasin, A. Surarerks and A. Rungsawang. “Web Spam Recognition by Edge Label”. In the proceeding of 14th Internation Annual Symposium on Computational Science and Engineering , Chiang Rai, Thailand March 23-26, 2010

[15] A. Arasu, J. Cho, H. Garcia-Molina, A. Paepcke and S. Raghavan. “Searching the Web”. ACM Transactions on Internet Technology, 1, 1, (August 2001), 2–43.

[16] Z. Gyongyi and H. Garcia-Molina. “Web Spam Taxomony”. First International Workshop on Adversarial Information Retrieval on the Web (AIRWeb 2005), May 10-14, 2005, Chiba, Japan.

[17] C. Castillo, D. Donato, L. Becchetti, P. Boldi, S. Leonardi, M. Santini, and S. Vigna. Reference Collection for Web Spam. ACM SIGIR Forum. (2006).

ประวัติผู้เขียนวิทยานิพนธ์

นายชาคริต ลิขิตขจร เกิดเมื่อวันที่ 27 ตุลาคม 2528 ที่จังหวัดกรุงเทพมหานคร สำเร็จการศึกษาปริญญา วิศวกรรมศาสตร์บัณฑิต ภาควิชาวิศวกรรมอุตสาหการ จุฬาลงกรณ์มหาวิทยาลัย ในปี พ.ศ. 2550 และได้เข้าศึกษาต่อในหลักสูตรวิศวกรรมศาสตร์มหาบัณฑิต คณะวิศวกรรมศาสตร์ ภาควิชาวิศวกรรมคอมพิวเตอร์ จุฬาลงกรณ์มหาวิทยาลัย ในปี พ.ศ. 2551