การสกัดความสัมพันธ์ระหว่างนิพจน์ระบุนามในภาษาไทย

โดย

นายรัฐภูมิ ตันสุตะพานิช

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต

สาขาวิชาวิทยาการคอมพิวเตอร์

ภาควิชาคอมพิวเตอร์

บัณฑิตวิทยาลัย มหาวิทยาลัยศิลปากร

ปีการศึกษา 2552

การสกัดความสัมพันธ์ระหว่างนิพจน์ระบุนามในภาษาไทย

โดย

นายรัฐภูมิ ตันสุตะพานิช

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต

สาขาวิชาวิทยาการคอมพิวเตอร์

ภาควิชาคอมพิวเตอร์

บัณฑิตวิทยาลัย มหาวิทยาลัยศิลปากร

ปีการศึกษา 2552

RELATION EXTRACTION AMONG NAMED ENTITIES IN THAI LANGUAGE

By

Rathapoome Tansutapanich

A Thesis Submitted in Partial Fulfillment of the Requirements for the Degree

MASTER OF SCIENCE

Department of Computing

Graduate School

SILPAKORN UNIVERSITY

2009

บัณฑิตวิทยาลัย มหาวิทยาลัยศิลปากร อนุมัติให้วิทยานิพนธ์เรื่อง “การสกัดความสัมพันธ์ระหว่างนิพจน์ระบุนามในภาษาไทย” เสนอโดย นายรัฐภูมิ ตันสุตะพานิช เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์

.....

(รองศาสตราจารย์ ดร.ศิริชัย ชินะตังกูร)

คณบดีบัณฑิตวิทยาลัย

วัน ..... เดือน ..... พ.ศ. .....

อาจารย์ที่ปรึกษาวิทยานิพนธ์

รองศาสตราจารย์ ดร.จันทนา ผ่องเพ็ญศรี

คณะกรรมการตรวจสอบวิทยานิพนธ์

.....ประธานกรรมการ

(อาจารย์ ดร.สุนีย์ พงษ์พินิจภิญโญ)

...../...../.....

.....กรรมการ

(ผู้ช่วยศาสตราจารย์ ดร.ณัฐธนนท์ หงส์วริทธิ์ธร)

...../...../.....

.....กรรมการ

(รองศาสตราจารย์ ดร.จันทนา ผ่องเพ็ญศรี)

...../...../.....

48307308 : สาขาวิชาวิทยาการคอมพิวเตอร์

คำสำคัญ : การประมวลผลภาษาธรรมชาติ / การสกัดความสัมพันธ์ / นิพจน์ระบุนาม / ฮิวริสติก

รัฐภูมิ ตันสุตะพานิช : การสกัดความสัมพันธ์ระหว่างนิพจน์ระบุนามในภาษาไทย. อาจารย์ที่ปรึกษาวิทยานิพนธ์ : รศ.ดร. จันทนา ผ่องเพ็ญศรี. 133 หน้า.

งานวิจัยนี้มีจุดมุ่งหมายที่จะพัฒนาการประมวลผลภาษาธรรมชาติในด้านของการค้นหาความสัมพันธ์ของนิพจน์ระบุนามในภาษาไทย โดยจะแบ่งการทำงานเป็น 2 ขั้นตอน คือ การสกัด NE (นิพจน์ระบุนาม) และการสกัดความสัมพันธ์

สำหรับในส่วนของการสกัด NE ได้ใช้วิธีการนำรายชื่อที่ได้แบ่งประเภทมาแล้ว ร่วมกับการใช้กฎฮิวริสติก เพื่อช่วยเพิ่มประสิทธิภาพในการสกัด NE สำหรับขั้นตอนของการสกัดความสัมพันธ์ โดยในขั้นตอนนี้จะใช้กฎฮิวริสติกในการสกัดความสัมพันธ์ โดยการสร้างคำสำคัญขึ้นมา ซึ่งคำสำคัญเหล่านี้สามารถบ่งบอกถึงความสัมพันธ์ที่อยู่ในข้อความได้ นอกจากนี้ยังมีการสร้างกฎแบบต่างๆ เพื่อให้ระบบสามารถเข้าใจลักษณะการเขียนข้อความแบบต่างๆ ได้

ในส่วนของการทดลองจะใช้ข้อความข่าวหรือบทความทางด้านของเทคโนโลยีคอมพิวเตอร์มาใช้ในการทดสอบทั้งในส่วนของการสกัด NE และการสกัดความสัมพันธ์ โดยจะใช้ข้อความทั้งหมด 300 ข้อความ และนำมาแบ่งเป็นสองส่วน คือ 100 ข้อความจะถูกใช้ในการฝึกฝนระบบ และอีก 200 ข้อความจะถูกใช้ในการทดสอบจริง โดยจะทำการสกัด NE ก่อนในขั้นแรก และจากนั้นจึงนำผลลัพธ์ของการสกัด NE ที่ได้ มาเข้าสู่ขั้นตอนของการสกัดความสัมพันธ์

ในการวัดประสิทธิภาพของระบบจะพิจารณาจากค่าระลึก (Recall), ค่าความแม่นยำ (Precision), และค่าอัตราการรู้จำ (F-measure) โดยผลของการสกัด NE คือ 88.82, 98.48, และ 93.40 ตามลำดับ และผลของการสกัดความสัมพันธ์คือ 81.33, 89.05, และ 85.02 ตามลำดับ

ภาควิชาคอมพิวเตอร์ บัณฑิตวิทยาลัย มหาวิทยาลัยศิลปากร ปีการศึกษา 2552

ลายมือชื่อนักศึกษา .....

ลายมือชื่ออาจารย์ที่ปรึกษาวิทยานิพนธ์ .....

48307308 : MAJOR : COMPUTER SCIENCE
KEY WORD : NATURAL LANGUAGE PROCESSING / RELATION EXTRACTION / NAMED ENTITIES / HEURISTIC
RATHAPOOME TANSUTAPANICH : RELATION EXTRACTION AMONG NAMED ENTITIES IN THAI LANGUAGE. THESIS ADVISOR : ASSOC. PROF. CHANTANA PHONGPENSRI, Ph.D.. 133 pp.

The thesis's purpose is to develop the field of natural language processing, with relation extraction of Thai language. It composes of two parts, which are NE (Named Entity) extraction, and relation extraction.

NE extraction identifies NEs in texts with categorized lists of names, together with heuristic rules (used to increase performance). Relation extraction is the heuristic-based system. The keywords were built to indicate the relation type. Moreover, the rules were built to help the system recognize different forms of message writing.

In the experiments 300 statements and articles in the field of computer technology, were used. The samples were divided into 2 parts: 100 statements for training the system, and 200 for the real test. The processes of the experiments are extracting NE, and extracting relation from the result of NE extraction.

The performances of the system considered are Recall, Precision, F-measure scores. The scores of NE extraction are 88.82, 98.48, and 93.40, respectively and the scores of relation extraction are 81.33, 89.05, and 85.02, respectively.

Department of Computing Graduate School, Silpakorn University Academic Year 2009
Student's signature .....
Thesis Advisor's signature .....

กิตติกรรมประกาศ

ผู้วิจัยต้องขอขอบพระคุณ รองศาสตราจารย์ ดร.จันทนา ผ่องเพ็ญศรี อาจารย์ที่ปรึกษาวิทยานิพนธ์ ที่ให้คำปรึกษา คำแนะนำ ตลอดจนแนวทางในการทำวิจัย ให้สามารถสำเร็จลุล่วงได้ด้วยดี

ขอขอบพระคุณ อาจารย์ ดร.สุนีย์ พงษ์พินิจภิญโญ ประธานกรรมการ และ ผู้ช่วยศาสตราจารย์ ดร.ณัฐธนนท์ หงส์วริทธิ์ธร ผู้ทรงคุณวุฒิและกรรมการ ที่กรุณาให้คำแนะนำและตรวจสอบวิทยานิพนธ์

ขอขอบพระคุณคุณพ่อ คุณแม่ และพี่ชาย ที่ได้คอยให้ความช่วยเหลือ เป็นกำลังใจ และคอยสนับสนุนมาโดยตลอด จนสามารถดำเนินการวิจัยจนสำเร็จลุล่วงได้ด้วยดี

ขอขอบคุณเพื่อนๆ พี่ๆ น้องๆ และทุกๆ คนที่มีส่วนช่วยเหลือแก่ผู้วิจัยไม่ว่าจะในด้านใดก็ตาม

สุดท้ายนี้ขอขอบพระคุณคณาจารย์ทุกท่านตั้งแต่อดีตจนถึงปัจจุบันที่ได้ประสิทธิ์ประสาทความรู้ให้แก่ผู้วิจัย จนทำให้สามารถนำความรู้ที่มีมาใช้ในการดำเนินการวิจัยจนสำเร็จเป็นวิทยานิพนธ์ฉบับนี้

สารบัญ

สารบัญตาราง

สารบัญภาพ

บทที่ 1

บทนำ

นิพจน์ระบุนาม (Named Entity) หรือในที่นี้ จะเรียกว่า NE คือ นิพจน์ที่ทำหน้าที่ระบุชี้เฉพาะถึงสิ่งใดๆ เช่น ชื่อบุคคล ชื่อองค์กร ชื่อสถานที่ รวมไปถึงนิพจน์แสดงวันเวลา ปริมาณเงิน และเปอร์เซ็นต์ (Chinchor 2008) ระบบสกัด NE ได้ถูกเสนอขึ้น และได้รับการกำหนดนิยามกรอบการทำงานในงานประชุมวิชาการ Message Understanding Conferrence (MUC) ในปี ค.ศ. 1995 โดยมีวัตถุประสงค์เพื่อให้เป็นระบบที่ทำหน้าที่ในการวิเคราะห์หา NE ในเอกสาร พร้อมทั้งระบุประเภทของ NE

ความเป็นมาและความสำคัญของปัญหา

การสกัด NE (Named Entity Extraction หรือ Named Entity Recognition) ถือได้ว่าเป็นขั้นตอนที่สำคัญขั้นตอนหนึ่งสำหรับการพัฒนาระบบประมวลผลเอกสาร โดยเฉพาะอย่างยิ่งสำหรับระบบต่างๆ ที่เกี่ยวข้องกับการเข้าถึงข้อมูล ตัวอย่างเช่น ระบบสกัดข้อสนเทศ (Information Extraction) โดยการสกัด NE ได้อย่างถูกต้อง เป็นสิ่งสำคัญที่ทำให้การทำงานในขั้นตอนต่างๆ ของระบบสกัดข้อสนเทศให้ประสบความสำเร็จ เช่น กระบวนการ template filling ซึ่งการระบุชื่อของสิ่งใดๆ ที่มีบทบาทในเฟรมแสดงความสัมพันธ์ (relational frame) เป็นสิ่งที่สำคัญมาก หรือในระบบค้นคืนเอกสาร ระบบสกัด NE สามารถช่วยให้ระบบค้นคืนเอกสารสามารถค้นคืนเอกสารได้ตรงใจผู้ใช้มากขึ้น โดยเฉพาะเมื่อผู้ใช้ต้องการค้นคืนเอกสารที่เกี่ยวข้องกับชื่อใดชื่อหนึ่ง เช่น หลีกเลี่ยงผลลัพธ์การค้นคืนข้อมูลที่เกี่ยวกับนายพิจิตร หากผู้ใช้ต้องการเอกสารที่เกี่ยวข้องกับจังหวัดพิจิตรเท่านั้น เป็นต้น หรือในระบบการแปลภาษา การสกัด NE จะช่วยให้การแปลภาษามีความถูกต้องมากขึ้น เช่น ป้องกันการแปลความหมายของ NE เช่น แปลชื่อ “นางแดง” ไปสู่ “Mrs. Red” แทนที่จะเป็น “Mrs. Dang”

นอกจากนี้การสกัดความสัมพันธ์ (Relation Extraction) จาก NE นั้นยังเป็นขั้นตอนที่จะช่วยให้การค้นคืนสารสนเทศ (Information Retrieval) หรือ การสกัดข้อสนเทศ นั้นเป็นไปอย่างมีประสิทธิภาพมากขึ้นอีกด้วย ตัวอย่างเช่น “ไอซีถูกใช้เป็นตัวควบคุมการทำงานในระบบไมโครคอมพิวเตอร์” จากประโยคนี้เราจะได้ NE ดังนี้ E1 = “ไอซี”, E2 = “ระบบไมโครคอมพิวเตอร์” จากการพิจารณาประโยคจะเห็นว่า E1 และ E2 นั้นมีความสัมพันธ์กันในรูปแบบที่ถูกติดตั้ง (located) ซึ่งประโยชน์จากการสกัดความสัมพันธ์ระหว่าง NE นี้เองจะช่วยให้ข้อมูลในเอกสารต่างๆ นั้นถูกจัดเก็บอย่างมีรูปแบบมากขึ้นซึ่งสามารถนำไปพัฒนาระบบการค้นคืนสารสนเทศ และการประมวลภาษาธรรมชาติแขนงอื่นได้อีกด้วย

วิทยานิพนธ์นี้จะแบ่งการทดลองออกเป็นสองส่วนหลักคือ การสกัด NE จากเอกสาร โดยผ่านกระบวนการการสกัด NE โดยการนำข้อมูลจากภายนอกร่วมกับการสร้างกฎ เพื่อแยกประเภทของ NE ออกมาเป็น 4 ประเภท คือ ORG, PER, LOC และ PRO (ชื่อองกรค์, ชื่อบุคคล, ชื่อสถานที่, และชื่อสิ่งของ) และจากนั้นจึงนำข้อมูลที่มีการแยกประเภท NE ออกมาแล้ว มาทำการสกัดหาความสัมพันธ์ระหว่าง NE โดยการค้นหาคำสำคัญที่อยู่ในบริบทภายใน NE ทั้งสอง ซึ่งชนิดของความสัมพันธ์ที่ต้องการหาในวิทยานิพนธ์นี้มีอยู่ 3 ชนิด คือ go_to, located_in และ create

วัตถุประสงค์ของการศึกษา

1. เพื่อศึกษาทฤษฎีประยุกต์ที่มีอยู่ ในส่วนของเทคนิคทางด้านการประมวลผลภาษาธรรมชาติ เพื่อพัฒนากระบวนการที่เหมาะสมสำหรับการวิเคราะห์และสกัดความสัมพันธ์ระหว่าง NE ภาษาไทย

2. เพื่อพัฒนาระบบวิเคราะห์และสกัดความสัมพันธ์ระหว่าง NE ในเอกสารภาษาไทยโดยใช้ทฤษฎีประยุกต์

ขอบเขตของการศึกษา

1. เนื่องจากคลังข้อมูล (Corpus) ที่นำมาใช้ในระบบนี้เป็นคลังข้อมูลที่มีเนื้อหาเฉพาะด้านวิทยาศาสตร์และเทคโนโลยีเท่านั้น ดังนั้นเอกสารที่นำมาใช้ในการทดสอบนั้นจำเป็นต้องมีเนื้อหาเฉพาะด้านวิทยาศาสตร์และเทคโนโลยีด้วยเท่านั้น

2. ระบบสกัดความสัมพันธ์ระหว่าง NE นี้พัฒนาขึ้นสำหรับประมวลผลเอกสารที่มีรูปแบบการใช้ภาษาที่ดี (well-style written) เช่น ไม่มีการสะกดผิด เป็นต้น

3. ระบบที่ใช้ในการทำวิทยานิพนธ์นี้มุ่งเน้นในการวิเคราะห์คำภาษาไทยเท่านั้น และจะไม่สนใจคำที่เขียนทับศัพท์ เช่น ดาต้า คอนเนค เป็นต้น ยกเว้นคำที่นิยมใช้กันโดยทั่วไป เช่น คอมพิวเตอร์ เว็บไซต์ เป็นต้น

4. ในการสกัดความสัมพันธ์นั้น ลักษณะของความสัมพันธ์จะเกิดขึ้นระหว่าง NE สองคำ คือ NE ทางด้านซ้ายและทางด้านขวาของบริบท

5. ในการวัดความถูกต้องของการหาความสัมพันธ์นั้นจะใช้ผลจากค่า Recall, Precision, และค่า F

คำจำกัดความที่ใช้ในการศึกษา

ความสัมพันธ์ go_to คือความสัมพันธ์ระหว่าง NE หนึ่งที่เดินทางไปยังอีก NE หนึ่ง

ความสัมพันธ์ create คือความสัมพันธ์ระหว่าง NE หนึ่งที่สร้างหรือประดิษฐ์ NE หนึ่ง

ความสัมพันธ์ located_in คือความสัมพันธ์ระหว่าง NE หนึ่งที่มีที่ตั้งอยู่ในอีก NE หนึ่ง

ค่า F (F-measure) คืออัตราการรู้จำ หรือค่าเฉลี่ยที่ให้ความสำคัญกับความแม่นยำและความความครบถ้วนเท่าๆ กัน โดยสามารถคำนวณได้จากสูตร (สุฤดี ฉัตรไตรมงคล 2548 : 5)

$$F = (2 \times Precision \times Recall) / (Precision + Recall)$$

ค่า Precision คือค่าความแม่นยำ ที่จะแสดงให้เห็นว่าระบบที่พัฒนาขึ้นมามีความแม่นยำเพียงใด โดยสามารถคำนวณได้จากสูตร (สุฤดี ฉัตรไตรมงคล 2548 : 5)

ความแม่นยำ(P) = (จำนวนคำตอบที่ถูกต้องที่ระบบค้นพบ × 100) / จำนวนคำตอบทั้งหมดที่ระบบเลือกขึ้นมา

ค่า Recall คือค่าระลึก หรือค่าความครบถ้วน ที่จะแสดงให้เห็นว่าเมื่อระบบได้ทำการดึงคำตอบออกมาแล้วมีความถูกต้องเพียงใด โดยสามารถคำนวนได้จากสูตร (สุฤดี ฉัตรไตรมงคล 2548 : 6)

ค่าระลึก(R) = (จำนวนคำตอบที่ถูกต้องที่ระบบค้นพบ × 100) / จำนวนคำตอบทั้งหมดที่ถูกต้องทั้งหมดในเอกสาร

NE ย่อมาจาก Named Entity หรือ “นิพจน์ระบุนาม” ในภาษาไทย

ENAMEX เป็น tag ชนิดหนึ่งใน SGML ที่ใช้แทน NE ประเภทชื่อเฉพาะทั่วไป

NE ชนิด LOC (Location) คือ NE ที่เกี่ยวกับสถานที่หรือพื้นที่ รวมไปถึงอาคารสถานที่ต่างๆ เช่น ประเทศ ภูเขา แม่น้ำ ตึก อาคาร ถนน สวนสาธารณะ เป็นต้น

NE ชนิด ORG คือ NE ที่เกี่ยวกับองค์กร หน่วยงาน บริษัท ห้าง ร้านต่างๆ หรือกลุ่มคนที่มากกว่าหนึ่งคนขึ้นไปที่รวมตัวกันและมีความสัมพันธ์ในรูปแบบขององค์กร

NE ชนิด PRO (Product) คือชื่อของสิ่งของหรือสินค้า(ทั้งในรูปแบบที่จับต้องได้ และจับต้องไม่ได้) ตัวอย่างเช่น รถยนต์, อาหาร, เสื้อผ้า, หุ้น, รวมไปถึงสินค้าที่อยู่ลักษณะของการบริการด้วย เช่น เที่ยวบิน หรือการเช่าสัญญาณ เป็นต้น

NE ชนิด PER คือ NE ที่เกี่ยวกับบุคคล หรือชื่อคน

NUMEX เป็น tag ชนิดหนึ่งใน SGML ที่ใช้แทน NE ประเภทจำนวน

POS ย่อมาจาก Part Of Speech หมายถึงการกำกับหน้าที่ของคำในประโยค เช่น คำนาม คำกริยา เป็นต้น

RE ย่อมาจาก Relation Extraction หมายถึงการสกัดความสัมพันธ์

TIMEX เป็น tag ชนิดหนึ่งใน SGML ที่ใช้แทน NE ประเภทวันเวลา

บทที่ 2

งานวิจัยที่เกี่ยวข้อง

การสกัด NE

งานวิจัยที่สกัด NE ด้วยการสร้างกฎโดยผู้เชี่ยวชาญ

Stevenson และ Gaizauskas (Stevenson and Gaizauskas 2000 : 290-295) ใช้วิธีการนำข้อมูลจากภายนอกมาช่วยในการสกัด NE ซึ่งในรายชื่อนั้นประกอบด้วย ชื่อองค์กร สถานที่ ชื่อคน คำนำหน้าชื่อต่างๆ ที่นิยมใช้ในการนำหน้าชื่อ เช่น Mister หรือ Lord เป็นต้น และคำที่มักใช้กับบริษัทหรือองค์กร เช่น Limited หรือ Incorporated เป็นต้น จากนั้นนำข้อมูลที่ต้องการสกัด NE มาผ่านกระบวนการกำกับหน้าที่คำ (POS) และการตัดคำ และจึงนำมาเปรียบเทียบกับคลังข้อมูล (Name Matching) เพื่อหา NE ซึ่ง NE ที่ต้องการหาในงานวิจัยนี้คือ สถานที่ (Location) องค์กร (Organization) และบุคคล (Person) ส่วน NE ประเภทตัวเลข วันที่ และจำนวนเงินนั้นไม่ได้มีการค้นหา ซึ่งผู้พัฒนาอ้างว่า NE เหล่านี้สามารถที่จะระบุได้ง่ายเนื่องจาก NE เหล่านี้มีลักษณะเฉพาะที่สังเกตได้ง่ายอยู่ในตัวเอง

Gaizauskas และคณะ (Gaizauskas et al. 1995 : 207-219) ได้อธิบายถึงระบบ LaSIE ซึ่งถูกพัฒนาโดยมหาวิทยาลัยเชฟฟิลด์ ระบบนี้รองรับปัญหาทั้งหมดที่กำหนดไว้ในการประชุม MUC-6 และในส่วนของปัญหาการสกัด NE นั้นในระบบนี้ใช้วิธีการจับคู่คำในประโยคกับรายชื่อที่จัดเตรียมเอาไว้ ซึ่งประเภทของ NE ที่ระบบนี้สามารถค้นหาได้นั้น คือ ชื่อองค์กร สถานที่ ชื่อบุคคล วันที่/เวลา และเงินตรา

ระบบ Nominator (Wacholder, Ravin and Choi 1997) มีการใช้กฎฮิวริสติกเข้ามาช่วยในการสกัด NE โดยระบบนี้ใช้ความรู้ 2 ประเภทในการตรวจหา NE ซึ่งได้แก่ ความรู้จากบริบท (Context) และความรู้ภายนอก (World Knowledge) โดยใช้บทความจาก Wall Street Journal ในการทดสอบ แต่ในงานนี้ผู้วิจัยไม่ได้กล่าวถึงประเภทของ NE ที่ระบบสามารถสกัดได้เอาไว้ชัดเจน

แต่จากตัวอย่างที่มีอยู่ในงานนี้คาดว่า NE ที่สกัดได้ในระบบนี้น่าจะเป็น NE ประเภท ชื่อองค์กร ชื่อสถานที่ และชื่อบุคคล

งานวิจัยที่ใช้วิธีการสกัด NE แบบอัตโนมัติโดยใช้สถิติ หรือเทคนิคการเรียนรู้

วิธีการสกัด NE จากการฝึกฝนระบบที่ได้รับความนิยมวิธีหนึ่ง คือ การใช้ Hidden Markov Model ตัวอย่างของระบบที่ใช้วิธีนี้คือ ระบบ Nymble (Bikel et al. 1997 : 194-201) ซึ่งใช้การสร้างแบบจำลองแบบไบแกรมแยกกันสำหรับสกัด NE ในแต่ละประเภท การทำนายประเภทของคำถัดไป จะขึ้นกับคำและประเภทของคำก่อนหน้า งานวิจัยนี้ใช้ข้อมูลจากลักษณะคำในการสกัด NE โดยพิจารณาคำเป็นเสมือนคู่ลำดับ ประกอบด้วยคำ (word) และคุณสมบัติของคำ (word feature) เช่น การขึ้นต้นด้วยอักษรพิมพ์ใหญ่ เป็นต้น ซึ่งชนิดของ NE ที่ระบบนี้สามารถสกัดได้ได้แก่ ชื่อองค์กร ชื่อบุคคล ชื่อสถานที่ เวลา วันที่ เปอร์เซ็นต์ และจำนวนเงิน

Maximum Entropy เป็นเทคนิคหนึ่งที่งานวิจัยหลายงานได้นำมาปรับใช้กับการสกัด NE คุณลักษณะเด่นของแบบจำลองนี้ คือเป็นแบบจำลองที่สามารถรวมเอาข้อมูลต่างๆ เช่น ข้อมูลจากบริบท ข้อมูลจากฐานความรู้ ฯลฯ มาใช้ในการประมาณความน่าจะเป็นของ NE ในแต่ละประเภทที่เกิดขึ้นในบริบทใดๆ ตัวอย่างของระบบที่ใช้เทคนิคนี้ ได้แก่ระบบ MENE (Borthwick et al. 1998) โดยใช้ข้อสนเทศจากหลายแหล่งในการสกัด NE ในภาษาอังกฤษ ซึ่งได้แก่ คุณลักษณะคำ เช่นการใช้อักษรพิมพ์ใหญ่, ตำแหน่งที่เกิดของคำในเอกสาร, การพิจารณาจากตัวคำ และข้อสนเทศจากการพิจารณาการปรากฏในคลังคำศัพท์ รวมทั้งข้อสนเทศที่ได้จากระบบภายนอก คือการนำผลลัพธ์การสกัด NE ที่ได้จากระบบอื่นมาใช้ฝึกฝนให้ระบบเรียนรู้ข้อผิดพลาดของระบบเหล่านั้น เพื่อเป็นการปรับปรุงประสิทธิภาพของระบบให้สูงขึ้น โดยในการทดสอบนั้นได้ใช้บทความเกี่ยวกับอุบัติเหตุทางเครื่องบินสำหรับชุดข้อความฝึกฝนระบบ (Training set) และบทความเกี่ยวกับจรวดและขีปนาวุธสำหรับชุดข้อความทดสอบจริง (Test set) ซึ่งชนิดของ NE ที่ระบบนี้สามารถสกัดได้ได้แก่ ชื่อองค์กร ชื่อบุคคล ชื่อสถานที่ เวลา วันที่ เปอร์เซ็นต์ และจำนวนเงิน

ต้นไม้ประกอบการตัดสินใจ (Decision Tree) เป็นอีกเทคนิคที่มีการนำมาใช้ในการสกัด NE ในงานของ Sekine และคณะ (Sekine, Grishman and Shinnou 1998 : 171-178) ได้ใช้ต้นไม้

ประกอบการตัดสินใจในการสกัด NE ในภาษาญี่ปุ่น โดยข้อสนเทศที่ใช้พิจารณาเพื่อสร้างต้นไม้ได้แก่ ชนิดของคำ ประเภทของตัวอักษร เช่น คาตาคานะ ฮิรากานะ คันจิ และข้อสนเทศจากพจนานุกรม อย่างไรก็ดี ต้นไม้ในงานวิจัยนี้จะมีลักษณะพิเศษ คือ ที่โหนดปลายสุด (leaf node) จะไม่เก็บเฉพาะคำกำกับ (tag) เช่น เป็นชื่อบุคคล ชื่อสถานที่ เป็นต้น ที่เป็นไปได้มากที่สุดเท่านั้น แต่จะเก็บความน่าจะเป็นของคำกำกับทั้งหมดที่เป็นไปได้สำหรับโหนดนั้น พร้อมกับค่าความน่าจะเป็นของแต่ละคำกำกับ เพื่อให้ระบบสามารถเลือกคำกำกับที่ให้ค่าความน่าจะเป็นโดยรวมทั้งประโยคสูงสุด โดยไม่ขัดแย้งกับคำกำกับข้างเคียง โดยในการทดสอบนั้นได้ใช้บทความเกี่ยวกับอุบัติเหตุทางยานพาหนะ และบทความเกี่ยวกับบุคคลผู้ประสบความสำเร็จในการทดสอบ ชนิดของ NE ที่ระบบนี้สามารถสกัดได้ได้แก่ ชื่อองค์กร ชื่อบุคคล ชื่อสถานที่ เวลา วันที่ เปอร์เซ็นต์ และจำนวนเงิน

งานวิจัยของ Collins และ Singer (Collins and Singer 1999 : 100-110) ใช้วิธีการแบบ Unsupervised ที่ไม่จำเป็นต้องใช้คลังข้อมูลจำนวนมากเหมือนแบบ Supervised โดยใช้กฎเริ่มต้นซึ่งเป็นกฎการสะกด (spelling rule) ทั้งหมด 7 ข้อ ซึ่งเป็นกฎที่พิจารณารูปคำ (lexical form) หรือองค์ประกอบภายในคำ โดยกฎเหล่านี้ จะเป็นเสมือนจุดเริ่มต้นในการเหนี่ยวนำกฎใหม่ ซึ่งเป็นทั้งกฎเชิงบริบทและกฎการสะกดต่อไป ส่วน Cucerzan และ Yarowsky (Cucerzan and Yarowsky 1999 : 90-99) ใช้วิธีการ Bootstrapping แบบ EM (Estimation-Maximization) ซึ่งมีพื้นฐานจากการเรียนรู้ซ้ำๆ และการประมาณค่าใหม่ (re-estimation) ของรูปแบบบริบทและรูปแบบโครงสร้าง หรือลักษณะการสะกดภายในหน่วยคำ ชนิดของ NE ที่ระบบนี้สามารถสกัดได้ได้แก่ ชื่อองค์กร ชื่อบุคคล ชื่อสถานที่

งานวิจัยที่ใช้แนวทางการสกัด NE แบบผสม

ระบบ LTG (Mikheev, Grover and Moens 1998 : 1-11) ซึ่งใช้กฎร่วมกับเทคนิคการเปรียบเทียบส่วนคำด้วยการใช้สถิติ (Statistical Partial Matching Technique) ระบบนี้ มีการทำงานเป็นลำดับขั้นตอนที่สอดแทรกสลับกัน โดยแต่ละขั้นตอน จะใช้ข้อสนเทศจากผลลัพธ์การสกัด NE ของขั้นตอนก่อนหน้าในการเปรียบเทียบกับคำหรือบางส่วนของคำ ซึ่งในงานนี้ ได้นำเอาแมกซิมั่มเอนโทรปี มาเป็นแบบจำลองทางสถิติเพื่อตรวจสอบความถูกต้องจากผลการเปรียบเทียบ ชนิดของ

NE ที่ระบบนี้สามารถสกัดได้ได้แก่ ชื่อองค์กร ชื่อบุคคล ชื่อสถานที่ เวลา วันที่ เปอร์เซ็นต์ และจำนวนเงิน

ที่ได้กล่าวมาทั้งหมดนั้นเป็นงานวิจัยด้านการสกัด NE ในภาษาต่างประเทศ ซึ่งได้นำมาแสดงอีกครั้งในตารางที่ 1 เพื่อช่วยในการเปรียบเทียบงานวิจัยต่างๆ

ตารางที่ 1 แสดงการเปรียบเทียบงานวิจัยต่างๆ ที่เกี่ยวกับการสกัด NE ในภาษาต่างประเทศ (เนื่องจากในบางงานวิจัยอาจมีค่าผลลัพธ์หลายค่า ซึ่งแยกตามการทดสอบแบบต่างๆ ดังนั้นค่าผลลัพธ์ที่แสดงในตารางจึงอาจเป็นค่าเฉลี่ย)

ผู้วิจัย	**วิธีการที่นำมาใช้**	**ชนิดของ NE ที่สกัดได้**	**ผลการสกัด**		
			Recall (%)	**Precision (%)**	**F (%)**
Stevenson และ Gaizauskas (Stevenson and Gaizauskas 2000 : 290-295)	ใช้รายชื่อจากภายนอก	LOC, ORG, PER	91	83	87
Gaizauskas และคณะ (Gaizauskas et al. 1995 : 207-219)	รวบรวมการวิเคราะห์ภาษาหลายชนิด เช่น การวิเคราะห์หน่วยคำ, การวิเคราะห์เชิงโครงสร้าง, การวิเคราะห์เชิงความหมาย	LOC, ORG, PER, Date, Time, Money	79	94	85.91
Wacholder และคณะ (Wacholder et al 1997)	กฎฮิวริสติก	LOC, ORG, PER	91	92	91.5
Bikel และคณะ (Bikel et al. 1997 : 194-201)	Hidden Markov Model และ กฎฮิวริสติก	LOC, ORG, PER, Date, Time, Money, Percentage	-	-	93

ตารางที่ 1 (ต่อ)

ผู้วิจัย	วิธีการที่นำมาใช้	ชนิดของ NE ที่สกัดได้	ผลการสกัด		
			Recall (%)	**Precision (%)**	**F (%)**
Borthwick และคณะ (Borthwick et al. 1998)	Maximum Entropy และกฎฮิวริสติก	LOC, ORG, PER, Date, Time, Money, Percentage	78	91	84.22
Sekine และคณะ (Sekine et al. 1998 : 171-178)	Dicision Tree	LOC, ORG, PER, Date, Time, Money, Percentage	-	-	84.5
Collins และ Singer (Collins and Singer 1999 : 100-110)	DL-CoTrain Algorithm, CoBoost Algorithm	LOC, ORG, PER	83.08	-	-
Cucerzan และ Yarowsky (Cucerzan and Yarowsky 1999 : 90-99)	Bootstrapping	LOC, ORG, PER	34.02	98.67	50.58
Mikheev และคณะ (Mikheev et al. 1998 : 1-11)	ใช้กฎร่วมกับเทคนิคการเปรียบเทียบส่วนคำด้วยการใช้สถิติ	LOC, ORG, PER, Date, Time, Money, Percentage	-	-	93.39

งานวิจัยที่เกี่ยวกับการสกัด NE ในภาษาไทย

Charoenpornsawat และคณะ (Charoenpornsawat, Kijsirikul and Meknavin 1998) เป็นการหาชื่อเฉพาะ (Proper Name) โดยใช้การพิจารณาคุณสมบัติ (feature-based approach) ซึ่งได้แก่ บริบท (Context) และการเกิดร่วมกันของชนิดคำ (Collocation) รวมทั้งใช้ฮิวริสติกในการสร้างชุดสมาชิกของชื่อเฉพาะที่เป็นไปได้ขึ้นมา และใช้แบบจำลองการเรียนรู้แบบวินโนว (Winnow

algorithm) สำหรับการระบุชื่อเฉพาะในภาษาไทยจากเอกสารที่ผ่านการตัดคำและกำกับชนิดของคำ ผลการทดลองของงานวิจัยนี้มีความถูกต้อง 92.17% สำหรับงานวิจัยนี้ไม่ได้กล่าวเอาไว้ว่าสามารถหาชื่อเฉพาะชนิดใดได้บ้าง

ต่อมา Kijsirikul และคณะ (Kijsirikul, Charoenpornsawat and Meknavin 1999) ได้นำอัลกอริธึมริปเปอร์ (Ripper) มาปรับใช้ในการสกัด NE ภาษาไทย เพื่อเปรียบเทียบประสิทธิภาพกับอัลกอริธึมวินโนว โดยผลการทดลองแสดงให้เห็นว่า อัลกอริธึมวินโนวมีประสิทธิภาพดีกว่าริปเปอร์ สำหรับงานวิจัยนี้ไม่ได้กล่าวเอาไว้ว่าสามารถหาชื่อเฉพาะชนิดใดได้บ้าง

Kawtrakul และคณะ (Kawtrakul et al. 1997) ได้เสนอวิธีในการรู้จำคำไม่รู้จักในภาษาไทย รวมถึงการรู้จำชื่อเฉพาะด้วย งานวิจัยนี้ ใช้โมเดลแบบผสมระหว่างแบบจำลองทางสถิติและการใช้กฎแบบขึ้นกับบริบท (context sensitive) โดยใช้โมเดลการกำกับความหมายของคำ (semantic tagging model) แทนโมเดลการกำกับด้วยชนิดของคำ ในการรู้จำคำไม่รู้จัก และใช้แบบจำลองทางสถิติในการระบุขอบเขตของคำไม่รู้จัก ผลลัพธ์ของระบบมีความถูกต้องประมาณ 85%

Chanlekha และ Kawtrakul (Chanlekha and Kawtrakul 2004) เสนอแนวทางการสกัด NE ภาษาไทย โดยใช้ Maximum Entropy Model ร่วมกับกฎและคลังคำศัพท์ รวมทั้งใช้สถิติของคำจากคลังเอกสารเพื่อหาตำแหน่ง ขอบเขต และประเภทของ NE จากการทดสอบประสิทธิภาพของระบบ โดยมุ่งเน้นไปที่การทดสอบกับข้อความประเภทข่าวเศรษฐกิจและการเมือง พบว่ามีค่าความถูกต้องเท่ากับ 85.29%, 82.67% และ 82.43% สำหรับชื่อบุคคล องค์กร และสถานที่ ตามลำดับ

สำหรับงานวิจัยด้านการสกัด NE ในภาษาไทยที่ได้กล่าวมานั้น ถูกนำมาแสดงอีกครั้งในตารางที่ 2 เพื่อช่วยในการเปรียบเทียบงานวิจัยต่างๆ

ตารางที่ 2 แสดงการเปรียบเทียบงานวิจัยต่างๆ ที่เกี่ยวกับการสกัด NE ในภาษาไทย (เนื่องจากในบางงานวิจัยอาจมีค่าผลลัพธ์หลายค่า ซึ่งแยกตามการทดสอบแบบต่างๆ ดังนั้นค่าผลลัพธ์ที่แสดงในตารางจึงอาจเป็นค่าเฉลี่ย)

ผู้วิจัย	วิธีการที่นำมาใช้	ชนิดของ NE ที่สกัดได้	ผลการสกัด		
			Recall (%)	Precision (%)	F (%)
Charoenpornsawat และคณะ (Charoenpornsawat et al. 1998)	Winnow Algorithm	ไม่ระบุ	-	-	92.17
Kijsirikul และคณะ (Kijsirikul et al. 1999)	Ripper Algorithm	ไม่ระบุ	85.94	96.43	90.88
Kawtrakul และคณะ (Kawtrakul et al. 1997)	วิธีการผสมระหว่างวิธีการทางสถิติและการสร้างกฎ	คำที่ไม่รู้จักในภาษาไทย	-	-	88.5
Chanlekha และ Kawtrakul (Chanlekha and Kawtrakul 2004)	Maximum Entropy, กฎฮิวริสติก	LOC, ORG, PER	88.62	88.49	88.55

การสกัดความสัมพันธ์ระหว่าง NE

งานวิจัยที่ใช้วิธีการเรียนรู้แบบ Supervised

งานของ Roth และ Yih (Roth and Yih 2002) ใช้การฝึกฝนระบบให้รู้จักจำแนก NE และความสัมพันธ์ก่อน และจึงใช้กฎความน่าจะเป็นในการสกัด NE และความสัมพันธ์ ต่อจากนั้นพวกเขาได้เสนอใช้การโปรแกรมเชิงเส้น (Linear Programming) (Roth and Yih 2004, 2007 : 1-26) เพื่อนำมาใช้ในการตัดสินใจเลือกข้อมูลที่จะนำมาเป็น NE หรือความสัมพันธ์ ซึ่งในงานวิจัยนี้ได้ใช้คลังข้อมูลที่มีการกำกับ NE และความสัมพันธ์ไว้แล้ว เพื่อช่วยในการเรียนรู้ของระบบ แต่ระบบของ Roth และ Yih นี้จะไม่ทำการกำหนดขอบเขตของ NE กล่าวคือ input ของระบบนี้จะต้องผ่าน

การกำหนดขอบเขตของ NE มาแล้ว และเมื่อนำเข้าสู่ระบบ ระบบก็จะทำการแยกประเภทของ NE และหาความสัมพันธ์ต่อไป โดยข้อความที่นำมาทดสอบนั้นจะใช้เอกสารจาก TREC ซึ่งจะประกอบด้วยข่าวจาก Wall Street Journal, Associate Press, และ San Jose Mercury News

Zelenko และคณะ (Zelenko, Aone and Richardella 2003) นั้นใช้วิธีการ Kernel Methods (KMs) ในการหาความสัมพันธ์ ร่วมกับระบบการแยกคำแบบ Shallow Parsing แต่งานวิจัยนี้ยังมีข้อผิดพลาดในการวิเคราะห์ประโยค งานวิจัยนี้ใช้ข้อมูลทั้งหมด 200 บทความที่เกี่ยวกับข่าวจากแหล่งต่างๆ เพื่อสกัด NE 3 ชนิดได้แก่ ชื่อบุคคล ชื่อองค์กร และชื่อสถานที่ และหาความสัมพันธ์ทั้งหมด 2 ชนิดคือ person-affiliation(ความสัมพันธ์ของบุคคลกับองค์กร) และ organization-location(องค์กรตั้งอยู่ที่) โดยใช้ Shallow Parser ในการวิเคราะห์ประโยค ซึ่งประโยคที่ไม่พบความผิดพลาดนั้นจะถูกเก็บเอาไว้ (90% จากประโยคทั้งหมด) และประโยคเหล่านี้จะถูกสุ่มออกมาเพื่อนำไปใช้ในการฝึกฝนระบบ (60%) และส่วนที่เหลือ (40%) จะถูกเอาไปใช้ในการทดสอบ โดยเนื้อหาของข้อความที่นำมาทดสอบนั้นคือข่าวจากสำนักข่าวต่างๆ ค่า F ของงานวิจัยนี้อยู่ที่ไม่น้อยกว่า 80%

Culotta และ Sorensen (Culotta and Sorensen 2004) ใช้ Kernel Methods เช่นเดียวกันแต่ใช้ Dependency Tree ในการวิเคราะห์ประโยคเพื่อลดข้อผิดพลาดในการวิเคราะห์ประโยค ในขั้นตอนของการฝึกฝนระบบนั้นมีการกำหนดความสัมพันธ์ทั้งหมด 24 ชนิด แต่ในการทดสอบนั้นได้ถูกจัดกลุ่มใหม่ให้เหลือ 5 ชนิดใหญ่ๆ เท่านั้น ได้แก่ AT(ตั้งอยู่ที่), NEAR(ใกล้), PART(ส่วนประกอบของ), ROLE(หน้าที่) และ SOCIAL(ความสัมพันธ์ทางสังคม) และ NE ที่ใช้ในงานวิจัยนี้มีทั้งหมด 5 ชนิด ได้แก่ ชื่อบุคคล, ชื่อองค์กร, GPE, ชื่อสถานที่ และชื่ออาคาร งานวิจัยนี้ได้ค่า Precision อยู่ที่ประมาณ 70.3% แต่ได้ Recall ที่ต่ำคือ 26.3% เท่านั้น

Giuliano และคณะ (Giuliano, Lavelli and Romano 2007 : 1-24) ใช้การประมวลผลภาษาแบบ Shallow ในการวิเคราะห์ประโยคซึ่งประกอบด้วย Tokenization, การแบ่งประโยค, การกำกับหน้าที่ของคำ และ Lemmatization โดยใช้ Kernel Methods ในการหาความสัมพันธ์ห้าชนิดจากข้อมูลที่เกี่ยวกับข่าว งานวิจัยนี้ได้เปรียบเทียบผลการทดลองกับงานของ Roth and Yih (2007) และจากผลการเปรียบเทียบนั้นงานของ Giuliano และคณะให้ผลที่ดีกว่า ชนิดของ NE ที่พวกเขา

ต้องการหาในงานนี้คือ ชื่อบุคคล, ชื่อสถานที่ และชื่อองค์กร และความสัมพันธ์ที่พวกเขาต้องการหาคือ work_for(ทำงานที่), located_in(ตั้งอยู่ใน), live_in(อาศัยอยู่ใน) และ kill(ฆ่า)

งานวิจัยที่ใช้วิธีการเรียนรู้แบบ Unsupervised

Brin (1998) ใช้วิธีการแบบ Bootstrapping ในการหาความสัมพันธ์ของหนังสือจากเว็บไซต์ โดยใช้ข้อมูลตั้งต้น (Initial Seed) เพียงเล็กน้อย (ชื่อผู้แต่ง และชื่อหนังสือ) จากตัวอย่างหนังสือ (ซึ่งใช้เพียงห้าเล่ม) จากนั้นหาข้อมูลทุกอย่างที่เกี่ยวข้องกับตัวอย่าง เพื่อสร้างรูปแบบ (Pattern) เพื่อใช้ในการอ้างอิง แล้วจึงนำไปใช้ในการค้นหาหนังสือเล่มใหม่ผ่านทางเว็บไซต์ ต่อมา Agichtein และ Gravano (Agichtein and Gravano 2000) ได้พัฒนาวิธีการของ Brin โดยนำการกำกับ NE มาใช้เพื่อช่วยเพิ่มประสิทธิภาพในการค้นหา โดยในงานนี้พวกเขาได้มุ่งค้นหาความสัมพันธ์เกี่ยวกับสถานที่ตั้งขององค์กร และข้อความที่นำมาใช้ทดสอบนั้นเป็นข้อความประเภทข่าว

Ravichandran และ Hovy (Ravichandran and Hovy 2002) ก็ได้ใช้วิธีการเดียวกับ Brin และ Agichtein โดยนำมาใช้กับระบบตอบคำถาม (Question Answering) แต่ถึงแม้จะใช้ข้อมูลตั้งต้นจำนวนน้อยเช่นเดียวกันแต่ในงานนี้กลับไม่ได้บอกชัดเจนว่าจะนำเอาข้อมูลนั้นไปใช้ได้โดยวิธีใดหรือใช้จำนวนเท่าใด

Lin และ Pantel (Lin and Pantel 2001 : 323-328) ใช้ Dependency Tree ช่วยในการพิจารณากริยาวลี (Verb Phrase) และประธานกับวัตถุในประโยคโดยใช้กฎว่า หากกริยาวลีใดมีประธานและวัตถุที่เหมือนกันก็จะอนุมานว่ามีความสัมพันธ์กัน แต่อย่างไรก็ตามวิธีการนี้ต้องการตัวอย่างกริยาวลีเพื่อใช้เป็นข้อมูลตั้งต้น

Hasegawa และคณะ (Hasegawa, Sekine and Grishman 2004) ได้เสนอวิธีการจับกลุ่มคู่ของ NE ร่วมกับความเหมือนกับของบริบทที่เกิดขึ้นระหว่าง NE ในการทดลองนั้นได้ใช้ข้อมูลจากหนังสือพิมพ์ปริมาณหนึ่งปี ได้ผลจากการทดลองว่าสามารถสกัดความสัมพันธ์ระหว่าง NE ได้ด้วยค่า Recall และ Precision ที่สูง (เฉลี่ยประมาณ 75-80%) แต่ในงานวิจัยนี้ไม่ได้กล่าวเอาไว้ว่าสามารถสกัดความสัมพันธ์ชนิดใดได้บ้าง

สำหรับงานวิจัยด้านการสกัดความสัมพันธ์ระหว่าง NE ในภาษาต่างประเทศ ถูกนำมาแสดงอีกครั้งในตารางที่ 3 เพื่อช่วยในการเปรียบเทียบงานวิจัยต่างๆ

ตารางที่ 3 แสดงการเปรียบเทียบงานวิจัยต่างๆ ที่เกี่ยวกับการสกัดความสัมพันธ์ในภาษาต่างประเทศ (เนื่องจากในบางงานวิจัยอาจมีค่าผลลัพธ์หลายค่า ซึ่งแยกตามการทดสอบแบบต่างๆ ดังนั้นค่าผลลัพธ์ที่แสดงในตารางจึงอาจเป็นค่าเฉลี่ย)

ผู้วิจัย	**วิธีการที่นำมาใช้**	**ชนิดของ** NE ที่สกัดได้	**ชนิดของความสัมพันธ์ที่สกัดได้**	**ผลการสกัด NE**			**ผลการสกัดความสัมพันธ์**		
				Rec (%)	**Prec (%)**	**F (%)**	**Rec (%)**	**Prec (%)**	**F (%)**
Roth และ Yih (Roth and Yih 2002)	กฎความน่าจะเป็น	PER, LOC	kill, born_in	88.17	85.15	86.63	62.34	75.18	68.16
Roth และ Yih (Roth and Yih 2007 : 1-26)	สูตร Linear Programming	PER, ORG, LOC	located_in, work_for, orgBased_in, live_in, kill	83.47	90.9	87.03	50.54	74.56	60.24
Zelenko และคณะ (Zelenko et al. 2003)	Kernel Methods, Shallow Parsing	PER, ORG, LOC	person-affiliation (ความสัมพันธ์ของบุคคลกับองค์กร), organization-location (องค์กรตั้งอยู่ที่)	-	-	-	80.03	91.55	85.05

ตารางที่ 3 (ต่อ)

ผู้วิจัย	วิธีการที่นำมาใช้	ชนิดของ NE ที่สกัดได้	ชนิดของความสัมพันธ์ที่สกัดได้	ผลการสกัด NE			ผลการสกัดความสัมพันธ์		
				Rec (%)	Prec (%)	F (%)	Rec (%)	Prec (%)	F (%)
Culotta และ Sorensen (Culotta and Sorensen 2004)	Kernel Methods, Dependency Tree	PER, ORG, GPE, LOC, FAC	at (ตั้งอยู่ที่), near (ใกล้), part (ส่วนประกอบของ), role (หน้าที่), social (ความสัมพันธ์ทางสังคม)	-	-	-	26.3	70.3	38
Giuliano และคณะ (Giuliano et al. 2007 : 1-24)	Shallow Processing, Kernel Methods	PER, ORG, LOC	work_for, located_in, live_in, kill	84.73	81.77	83.17	49.26	66.45	56.58
Agichtein และ Gravano (Agichtein and Gravano 2000)	Bootstrapping	ORG, LOC	organization-location (องค์กรตั้งอยู่ที่)	-	-	-	45	76	56.53

ตารางที่ 3 (ต่อ)

ผู้วิจัย	วิธีการที่นำมาใช้	ชนิดของ NE ที่สกัดได้	ชนิดของความสัมพันธ์ที่สกัดได้	ผลการสกัด NE			ผลการสกัดความสัมพันธ์		
				Rec (%)	Prec (%)	F (%)	Rec (%)	Prec (%)	F (%)
Hasegawa และคณะ (Hasegawa, Sekine and Grishman 2004)	การวิเคราะห์ข้อความภายในบริบท และประเภทของคู่ NE	PER, GPE, ORG	affiliation, role, location, part-whole, social relationship				78.5	77.5	77.5

สำหรับการสกัดความสัมพันธ์ระหว่าง NE ในภาษาไทยนั้น จากที่ได้ทำการสืบค้นแล้วยังไม่พบงานวิจัยทางด้านนี้

บทที่ 3

ทฤษฎีที่เกี่ยวข้อง

ในวิทยานิพนธ์นี้มีการนำทฤษฎีต่างๆ ที่เกี่ยวข้องกับการวิจัยในครั้งนี้ ดังนี้

ลักษณะของ NE

ในการประชุมวิชาการ Message Understanding Conferences ครั้งที่ 6 (MUC-6) (MUC-6 1996) นั้น ได้มีการกำหนดกรอบนิยามของ NE เอาไว้ (Named Entity Task Definition 1995) ซึ่งปัญหาของ NE นั้นถูกแบ่งออกได้เป็น 3 ปัญหาย่อยๆ คือ

1. การหาชื่อเฉพาะทั่วไป (Entity Name) ประกอบด้วย ชื่อองค์กร (Organization) ชื่อบุคคล (Person) และ ชื่อสถานที่ (Location)

2. การหาข้อความแสดงวันเวลา (Temporal Expression) ประกอบด้วย วันที่ (Date) และ เวลา (Time)

3. การหาข้อความแสดงจำนวน (Number Expression) ประกอบด้วย จำนวนเงิน (Monetary Value) และ เปอร์เซ็นต์ (Percentage)

ปัญหาทั้งสามประเภทของ NE นี้สามารถแทนค่าด้วยแท็ก SGML ดังนี้ ENAMEX, TIMEX และ NUMEX ตามลำดับ ซึ่งเมื่อต้องการระบุชนิดของ NE แต่ละประเภทก็สามารถกำหนดด้วย Attribute ที่เรียกว่า TYPE ซึ่งรูปแบบในการกำหนดชนิดของ NE จะมีลักษณะดังนี้ <ELEMENT-NAME ATTR-NAME="ATTR-VALUE" ...>text-string</ELEMENT-NAME> ตัวอย่างเช่น “<ENAMEX TYPE="ORGANIZATION">Taga Co.</ENAMEX>”

นิยามความหมายของ NE แต่ละชนิด

1. ชื่อเฉพาะทั่วไป (Entity Name) ใช้กับชื่อเฉพาะต่างๆ และชื่อย่อ แบ่งเป็น 3 ชนิด* คือ

1.1. องค์กร (Organization - ORG) ประกอบด้วย ชื่อบริษัท หน่วยงานรัฐบาล และองค์กรต่างๆ

1.2. บุคคล (Person - PER) ประกอบด้วย ชื่อบุคคล หรือนามสกุล

1.3. สถานที่ (Location - LOC) ประกอบด้วย ชื่อสถานที่ และลักษณะภูมิประเทศ (เมือง จังหวัด ประเทศ แม่น้ำลำคลอง ภูเขา เป็นต้น)

ในแผนการทดลองของ ACE05 (The ACE 2005 (ACE05) Evaluation Plan 2005 : 1) ได้เสนอชนิดของ NE เพิ่มเติมเข้ามา ดังนี้

1.4. สิ่งก่อสร้าง (Facility – FAC) สถานที่ที่เป็นสิ่งที่มนุษย์สร้างขึ้น ยกตัวอย่างเช่น สนามบิน สะพาน อาคารต่างๆ เป็นต้น

1.5. GPE (Geo-Political Entity) มีลักษณะคล้ายกับ Location แต่จะต่างกันตรงที่ GPE นี้จะมีลักษณะทางการเมืองหรือการปกครองเข้ามาเกี่ยวข้อง ยกตัวอย่างเช่น ทวีป ประเทศ เขตการปกครอง เป็นต้น

1.6. ยานพาหนะ (Vehicle – VEH) ยานพาหนะทั้งหมด ทั้งทางบก เรือ อากาศ

1.7. อาวุธ (Weapon – WEA) อาวุธต่าง เช่น อาวุธเคมี ระเบิด นิวเคลียร์ จรวด ของมีคม ปืน เป็นต้น

2. วันเวลา (Temporal Expression) ใช้กับข้อมูลที่เกี่ยวกับวันที่ หรือเวลา แบ่งเป็น 2 ชนิด** คือ

---

* ดูตัวอย่างการใช้งานได้ที่ http://www.cs.nyu.edu/cs/faculty/grishman/NEtask20.book_6.html#HEADING17

2.1. วันที่ (Date) ข้อมูลที่บอกถึงวัน (รวมไปถึง เดือน ปี ฤดูกาล)

2.2. เวลา (Time) ข้อมูลที่บอกถึงช่วงเวลา

3. จำนวน (Number Expression) ใช้กับจำนวนเงิน หรือเปอร์เซ็นต์ ซึ่งสามารถใช้ได้กับข้อมูลที่อยู่ในรูปตัวเลข หรือตัวอักษร แบ่งเป็น 2 ประเภท* คือ

3.1. จำนวนเงิน (Money) ข้อมูลที่บอกถึงจำนวนเงิน

3.2. เปอร์เซ็นต์ (Percent) ข้อมูลที่อยู่ในรูปของเปอร์เซ็นต์

ในงานของ Sekine (Sekine 2003) ได้เสนอชนิดของ NE เพิ่มเติมโดย NE ของ Sekine นั้นมีโครงสร้างเป็นลำดับขั้น (hierarchy) โดยที่ลำดับบนสุดนั้นยังมีแบ่งเป็น 3 ประเภทเหมือนที่ผ่านมา คือ ชื่อเฉพาะ(NAME), วันเวลา(TIME TOP), และจำนวน(NUMEX) ซึ่งในแต่ละประเภทนั้นยังสามารถแบ่งแยกย่อยออกเป็น NE ชนิดย่อยๆ ได้อีก ซึ่งล่าสุด NE ที่ Sekine เสนอเอาไว้มีทั้งหมด 200 ชนิด (Sekine 2007) สำหรับในส่วนของประเภทชื่อเฉพาะนั้น Sekine ได้แบ่งออกเป็นชนิดหลักๆ ดังนี้ (Sekine 2003)

1. Person คือชื่อคน

2. Organization ชื่อองค์กรที่ประกอบไปด้วยคนมากกว่าหนึ่งคน และยังรวมไปถึงชื่อของกลุ่มคน ตัวอย่างเช่น วงดนตรี อีกด้วย (ในส่วนนี้จะต่างจาก ACE ตรงที่กลุ่มคนนั้นจะจัดอยู่ในประเภทชื่อบุคคล)

3. Location ชื่อสถานที่ต่างๆ ที่ไม่ใช่สิ่งก่อสร้าง เช่น ประเทศ, ภูเขา, แม่น้ำ เป็นต้น

4. Facility คือสถานที่ที่มีลักษณะเป็นสิ่งปลูกสร้าง เช่น ตึก, อาคาร, ถนน, สวนสาธารณะ เป็นต้น

** ดูตัวอย่างการใช้งานได้ที่ http://www.cs.nyu.edu/cs/faculty/grishman/NEtask20.book_11.html#HEADING38

* ดูตัวอย่างการใช้งานได้ที่ http://www.cs.nyu.edu/cs/faculty/grishman/NEtask20.book_17.html#HEADING44

5. Product คือชื่อของสิ่งของหรือสินค้า(ทั้งในรูปแบบที่จับต้องได้ และจับต้องไม่ได้) ตัวอย่างเช่น รถยนต์, อาหาร, เสื้อผ้า, หุ้น, รวมไปถึงสินค้าที่อยู่ลักษณะของการบริการด้วย เช่น เที่ยวบิน หรือการเช่าสัญญาณ เป็นต้น

6. Event คือชื่อของเหตุการณ์ต่างๆ

7. Natural Object คือชื่อของสิ่งที่เกิดขึ้นตามธรรมชาติ เช่น โปรตอน, แบคทีเรีย, สัตว์, ต้นไม้ เป็นต้น

8. Title คำนำหน้าชื่อ หรือตำแหน่งของบุคคล

9. Unit หน่วยของสิ่งของต่างๆ เช่น กรัม, กิโลกรัม เป็นต้น รวมไปถึงหน่วยของเงิน เช่น บาท, เยน, ยูโร เป็นต้น

10. Vocation อาชีพ

11. Disease ชื่อของโรคหรืออาการบาดเจ็บต่างๆ

12. God ชื่อของพระเจ้า หรือเทพเจ้า เทวดา นางฟ้า เช่น Zeus, Venus เป็นต้น

13. ID Number เลขหรือรหัสที่ระบุถึงสิ่งต่างๆ เช่น เลขบัตรประชาชน, เลขพาสปอร์ต เป็นต้น

14. Color ชื่อของสี

สำหรับในวิทยานิพนธ์นี้จะทำการหา NE ทั้งหมด 4 ชนิดโดยจะอ้างอิงตามนิยามของ Sekine (Sekine 2003) ซึ่งได้แก่ ORG(Organization), PER(Person), LOC(Location), PRO(Product) โดยในส่วนของ LOC นั้นจะรวมเอาคุณสมบัติของ NE ชนิด Facility เข้าไปด้วย ทั้งนี้เพราะวิทยานิพนธ์นี้ไม่ได้ใช้ประโยชน์จากความแตกต่างของ NE ชนิด Location และ Facility

อย่างไรก็ตาม NE ในกลุ่มแรก คือ กลุ่มชื่อเฉพาะทั่วไป มักก่อให้เกิดปัญหาต่อระบบประมวลผลเอกสารมากกว่าการสกัด NE ในสองกลุ่มหลัง (Baluja, Mittal and Sukthankar 2000 : 1-2, quoting Palmer and Day 1997 : 190-193) เนื่องจากมักมีรูปแบบ และลักษณะการเกิดที่คลุมเครือมากกว่า ดังนั้นงานวิจัยเกี่ยวกับการวิเคราะห์และดึง NE โดยส่วนใหญ่ จึงมุ่งเน้นมาที่การหาชื่อเฉพาะทั่วไป แนวทางของการสกัด NE อาจแบ่งเป็นกลุ่มหลักๆ 3 กลุ่ม ได้แก่ แนวทางการใช้กฎ

หรือฐานความรู้ชื่อที่สร้างขึ้นโดยผู้เชี่ยวชาญ แนวทางแบบอัตโนมัติโดยใช้สถิติ หรือหลักการฝึกฝนระบบ และแนวทางแบบผสม

รูปแบบของ NE ภาษาไทย (Chanlekha and Kawtrakul 2004)

- NE ภาษาไทยไม่มีการใช้อักขระหรือเครื่องหมายพิเศษ ที่ช่วยบอกความแตกต่างระหว่าง NE และคำทั่วไปอย่างเช่น การใช้อักษรพิมพ์ใหญ่ในภาษาอังกฤษ
- NE ภาษาไทยไม่มีกฎเกณฑ์ในการสร้างที่แน่นอนกล่าวคือ สามารถสร้างขึ้นจากคำใดก็ได้ ทำให้ปริมาณ NE ภาษาไทยโตได้อย่างไม่มีขอบเขตจำกัด ส่งผลให้เกิดปัญหาในการรวบรวมคลังชื่อ
- NE ภาษาไทยมีทั้งแบบคำเดี่ยวและประกอบจากหลายคำเรียงกัน โดยเกิดขึ้นจากการรวมกันของคำ ทั้งคำที่รู้จักและไม่รู้จัก ดังตารางที่ 4

ตารางที่ 4 ลักษณะของ NE

รูปแบบ	**ตัวอย่าง**
คำไม่รู้จักทั้งหมด	เป๊ปซี่
คำรู้จักและไม่รู้จัก	**แม่**ฮ่อง**สอน**
คำรู้จักทั้งหมด	การไฟฟ้านครหลวง

ความสัมพันธ์ระหว่าง NE

แนวคิดของการสกัดความสัมพันธ์ (Relation Extraction – RE) (MUC-6 1996) ได้ถูกแนะนำในการประชุมวิชาการ Sixth Message Understanding Conference (MUC-6) เมื่อปี 1995 หลังจากนั้น MUC-7 (Chinchor 2008) ได้กำหนดแม่แบบปัญหาความสัมพันธ์ (Template Relation Task) ขึ้นพร้อมกับแบ่งชนิดของความสัมพันธ์ออกเป็นสามชนิด (employee_of, product_of, location_of) เพื่อใช้กับ NE ประเภทองค์กร หลังจากนั้นที่การประชุม Automatic Content

Extraction (ACE) ได้เสนอปัญหาการตรวจจับและอธิบายลักษณะความสัมพันธ์ (Relation Detection and Characterization (RDC)) ขึ้นในปี 2002 (Hasegawa, Sekine and Grishman 2004) ซึ่งงานส่วนใหญ่ของ ACE นั้นจะใช้วิธีการเรียนรู้ (Machine Learning) แบบ Supervised ซึ่งวิธีการเรียนรู้ นอกจากจะถูกใช้ในการสกัด NE แล้ว ในการสกัดความสัมพันธ์นั้นก็ยังนิยมใช้กันอย่างกว้างขวางอีกด้วย (Giuliano, Lavelli and Romano 2007 : 2) และวิธีการที่นำมาใช้นั้นมักจะเป็นระบบการเรียนรู้แบบ Supervised ยกตัวอย่างเช่น Kernel Methods (KMs), Maximum Entropy (ME), Hidden Markov Model (HMM) และ Conditional Random Field (CRF)

แต่วิธีการเรียนรู้แบบ Supervised นั้นจำเป็นต้องใช้คลังข้อมูลที่มีขนาดของข้อมูลที่ค่อนข้างใหญ่ และข้อมูลนั้นจะต้องถูกกำกับ NE หรือความสัมพันธ์ไว้แล้ว ซึ่งปัญหาของการใช้ข้อมูลลักษณะนี้คือจำเป็นต้องใช้เวลาและกำลังในการเตรียมข้อมูลเป็นอย่างมาก มีงานวิจัยบางงานที่ไม่ได้ใช้วิธีการแบบ Supervise (หรือใช้เพียงเล็กน้อย) ซึ่งข้อดีของวิธีนี้คือไม่ต้องการคลังข้อมูลที่มีขนาดใหญ่

วิธีการในการสกัด NE และความสัมพันธ์

การสร้างกฎโดยผู้เชี่ยวชาญ

เป็นระบบที่สร้างขึ้นโดยให้ผู้เชี่ยวชาญเป็นผู้วิเคราะห์และสร้างกฎสำหรับสกัด NE และความสัมพันธ์ ระบบในกลุ่มนี้มักมีพื้นฐานการทำงาน 3 วิธี (Baluja et al. 2000 : 1-2) ได้แก่

1. การใช้กฎในลักษณะของ Regular Expression ที่สร้างด้วยผู้เชี่ยวชาญ อย่างเช่นการใช้ Regular Expression ในการค้นหาคำสำคัญในประโยค

2. การใช้ข้อมูลหรือทรัพยากรเพิ่มเติมจากภายนอก เช่นตำแหน่งทางภูมิศาสตร์ รายชื่อบริษัท รายชื่อบุคคล เป็นต้น และใช้วิธีการระหว่างคำในเอกสารกับข้อมูลหรือทรัพยากรที่มี

3. การใช้กฎต่างๆ ทางภาษาศาสตร์ รวมไปถึงกฎ Heuristic สำหรับกฎ Heuristic คือวิธีที่ทำการทดลองค้นคว้าหากฎโดยใช้ วิจารญาณ หรือการลองผิดลองถูกของผู้ตัดสินใจ มีการเรียนรู้การตัดสินใจและการประเมินผล สามารถประยุกต์ใช้ได้หลากหลาย และวิธีการ Heuristic นี้จะใช้

กับระบบมีความซับซ้อนมากๆ จะช่วยลดเวลาในการทำงาน แต่รับประกันไม่ได้ว่าผลลัพธ์ที่ออกมาเป็นผลลัพธ์ที่ถูกต้องที่สุด

สำหรับการนำกฎ Heuristic มาใช้ในการวิเคราะห์ประโยคจะเป็นลักษณะการใช้พื้นฐานจากรูปแบบการใช้อักษรพิมพ์ใหญ่ เครื่องหมายวรรคตอน และตำแหน่งของ NE ในเอกสาร ยกตัวอย่างเช่นระบบ Nominator (Wacholder, Ravin and Choi 1997) ได้นำเอาความรู้จากกฎ Heuristic ด้วยการใช้ฐานความรู้ซึ่งประกอบด้วยข้อมูลพิเศษเกี่ยวกับชื่อ และลักษณะรูปคำ (lexical feature) ที่เกี่ยวข้อง ข้อมูลเหล่านี้ได้แก่ คำนำหน้าชื่อคน คำระบุนัยขององค์กร และชื่อสถานที่ รวมทั้งชุดคำยกเว้น (exception word) เช่น คำที่ขึ้นต้นด้วยอักษรพิมพ์ใหญ่ที่ไม่น่าจะเป็นชื่อแบบคำเดี่ยว (single word proper name) และยังได้นำเอาข้อสังเกตจากลักษณะการเขียนมาใช้ประโยชน์ด้วย นั่นคือ สำหรับวิธีการเขียนโดยทั่วไปนั้น เมื่อมีการอ้างอิงถึงชื่อใดๆ เป็นครั้งแรก มักจะอ้างอิงโดยใช้ชื่อเต็มของสิ่งนั้น ในขณะที่การอ้างอิงครั้งต่อๆไป อาจเป็นการเรียกแบบย่อซึ่งมีความคลุมเครือมากกว่า การใช้พื้นฐานจากรูปแบบการใช้อักษรพิมพ์ใหญ่ เครื่องหมายวรรคตอน และตำแหน่งของ NE ในเอกสาร ระบบนี้มีการใช้ฐานความรู้ซึ่งประกอบด้วยข้อมูลพิเศษเกี่ยวกับชื่อและลักษณะรูปคำ (lexical feature) ที่เกี่ยวข้อง ข้อมูลเหล่านี้ได้แก่ คำนำหน้าชื่อคน คำระบุนัยขององค์กร และชื่อสถานที่ รวมทั้งชุดคำยกเว้น (exception word) เช่น คำที่ขึ้นต้นด้วยอักษรพิมพ์ใหญ่ที่ไม่น่าจะเป็นชื่อแบบคำเดี่ยว (single word proper name) และยังได้นำเอาข้อสังเกตจากลักษณะการเขียนมาใช้ประโยชน์ด้วย นั่นคือ สำหรับวิธีการเขียนโดยทั่วไปนั้น เมื่อมีการอ้างอิงถึงชื่อใดๆ เป็นครั้งแรก มักจะอ้างอิงโดยใช้ชื่อเต็มของสิ่งนั้น ในขณะที่การอ้างอิงครั้งต่อๆไป อาจเป็นการเรียกแบบย่อซึ่งมีความคลุมเครือมากกว่า จากข้อเท็จจริงนี้ในระบบ Nominator จึงระบุตำแหน่งชื่อของสิ่งใดๆ ที่มีการอ้างอิงแบบเต็มรูปแบบ (full form) ก่อน จากนั้นจึงนำชื่อเต็มที่สกัดได้ไปใช้ในการสกัดชื่อของสิ่งนั้นที่ถูกอ้างอิงในรูปแบบที่สั้นหรือคลุมเครือกว่าในตำแหน่งอื่นๆในเอกสารต่อไป

ถึงแม้ระบบที่ใช้ผู้เชี่ยวชาญสร้างกฎนั้น มักให้ผลลัพธ์ที่มีความถูกต้องสูง และสามารถนำเอาความรู้ทางภาษาศาสตร์ หรือความรู้พิเศษจากผู้เชี่ยวชาญในแต่ละโดเมนมาใช้ในระบบได้โดยตรง อย่างไรก็ตาม แนวทางการทำงานดังกล่าวต่างก็มีข้อจำกัด เช่น วิธีการใช้ฐานข้อมูล ก็จะต้องใช้ฐานข้อมูลขนาดใหญ่ เพื่อให้สามารถเก็บชื่อเฉพาะต่างๆได้อย่างครอบคลุม ซึ่งอาจส่งผล

ให้เกิดปัญหาการควบคุมขนาดคลังคำศัพท์ได้ เนื่องจาก NE ใหม่เกิดเพิ่มขึ้นตลอดเวลา รวมทั้งวิธีนี้ยังยากต่อการวิเคราะห์ประเภทของ NE เช่น พิจิตร ซึ่งสามารถเป็นได้ทั้งชื่อบุคคล และชื่อสถานที่ เป็นต้น สำหรับวิธีการใช้กฎเพื่อการวิเคราะห์นั้น มีข้อด้อยคือ ระบบที่พัฒนาขึ้นโดยวิธีนี้ มักทำงานได้ดีเฉพาะในสถานการณ์หรือในโดเมนที่ผู้พัฒนาสนใจเท่านั้น เมื่อต้องการเปลี่ยนไปวิเคราะห์ภาษาอื่น หรือเปลี่ยนโดเมน ก็จะต้องสร้างกฎขึ้นใหม่ ซึ่งจากการทดลองของ Appelt และคณะ (Appelt et al. 1993) ได้แสดงให้เห็นว่า การใช้กฎไวยากรณ์ทั่วไปในการสกัด NE จะให้ผลลัพธ์ที่มีความถูกต้องต่ำกว่าระบบที่มีการปรับให้เหมาะสมกับแต่ละแอปพลิเคชั่น (application dependent) รวมทั้งเอกสารแต่ละประเภท ก็มักจะมีสำนวนและรูปแบบการเขียนแตกต่างกันไป กฎที่สามารถรู้จำรูปแบบพิเศษของเอกสารสามารถเพิ่มประสิทธิภาพให้กับระบบได้อย่างมาก เมื่อใช้กับเอกสารในโดเมนนั้น แต่อาจจะไม่สามารถทำงานได้ดีเมื่อนำไปใช้กับโดเมนอื่น จากการวัดประสิทธิภาพของระบบที่ใช้แนวทางการใช้กฎ มักแสดงให้เห็นว่า ประสิทธิภาพของระบบที่ลดลงนั้นเป็นผลสืบเนื่องมาจากการเปลี่ยนรูปแบบของเอกสาร และการเปลี่ยนโดเมนของเอกสารที่ใช้ทดลอง นอกจากนี้ ระบบที่สร้างกฎโดยผู้เชี่ยวชาญ ต้องใช้เวลาและแรงงานจากนักภาษาศาสตร์ในการพัฒนาระบบ จึงทำให้เสียเวลามากเมื่อต้องการนำระบบไปใช้กับภาษาอื่นหรือโดเมนอื่น รวมทั้งการสร้างกฎให้ครอบคลุม ยังเป็นกระบวนการที่ยาก ต้องใช้ความรู้ความชำนาญจากผู้เชี่ยวชาญ รวมทั้งอาจต้องเกี่ยวข้องกับการประมวลผลทางภาษาศาสตร์ที่ซับซ้อน เช่น การแจงประโยค เป็นต้น

แนวทางแบบอัตโนมัติโดยใช้สถิติ หรือเทคนิคการเรียนรู้

แนวทางการสกัด NE แบบอัตโนมัติโดยใช้วิธีการเรียนรู้ (machine learning) มีขึ้นเพื่อให้สามารถลดเวลาและแรงงานของนักภาษาศาสตร์ที่ต้องใช้ในการพัฒนาระบบ รวมทั้งลดเวลาและแรงงานในการปรับระบบไปใช้ในโดเมนหรือภาษาใหม่ เนื่องจากเมื่อต้องการเปลี่ยนระบบไปสู่โดเมนใหม่ ก็สามารถทำได้ด้วยการนำโปรแกรมเรียนรู้ ไปประมวลผลบนคลังเอกสารในโดเมนใหม่เพื่อฝึกฝนระบบให้สามารถเรียนรู้ได้ด้วยตัวเอง นอกจากนี้ การใช้เทคนิคการเรียนรู้ หรือแนวทางสถิติ ยังเป็นการสร้างความรู้จากคลังเอกสารจริงๆ โดยไม่ต้องอาศัยความรู้ความชำนาญจากนักภาษาศาสตร์ในการสร้างกฎ ข้อเด่นอีกประการหนึ่งของระบบที่ใช้แนวทางนี้คือ ผลลัพธ์

แบบค่าความน่าจะเป็นที่ได้จากแบบจำลองทางสถิตินั้น มีประโยชน์มากกว่าคำตอบแบบ “ใช่” หรือ “ไม่ใช่” จากระบบที่ใช้กฎและฐานความรู้ เนื่องจากให้ข้อมูลมากกว่า และสามารถนำไปใช้ได้หลากหลายกว่า

แนวทางในการวิเคราะห์และดึง NE โดยการสอนระบบให้เรียนรู้ที่นำมาใช้อย่างกว้างขวางโดยส่วนใหญ่ มักเป็นการฝึกฝนระบบแบบ Supervised ซึ่งต้องการตัวอย่างที่ใช้ฝึกฝนระบบเป็นจำนวนมาก เนื่องจากจำนวนเอกสารที่ใช้ในการฝึกฝนมีผลต่อประสิทธิภาพของระบบ โดยเมื่อเพิ่มจำนวนเอกสารในการสอน ความถูกต้องในการสกัด NE หรือความสัมพันธ์ มักจะมีความถูกต้องมากขึ้น อย่างไรก็ดี คลังเอกสารที่มีการกำกับเพื่อใช้ฝึกฝนระบบนั้น ไม่ได้มีพร้อมให้ใช้สำหรับทุกภาษาหรือทุกโดเมน รวมทั้งการสร้างคลังเอกสารเพื่อใช้ฝึกฝนระบบก็ต้องใช้แรงงานและเวลา นอกจากนี้ ความถูกต้องของการกำกับเอกสารที่ใช้ฝึกฝนระบบก็มีผลอย่างมากต่อประสิทธิภาพของระบบ เพื่อหลีกเลี่ยงปัญหาของระบบเรียนรู้ Supervised ที่ต้องการคลังเอกสารที่มีการกำกับคำขนาดใหญ่เพื่อฝึกสอนระบบ จึงได้มีงานวิจัยที่ใช้วิธีการแบบ Unsupervised เพื่อให้มีการใช้คลังข้อมูลให้น้อยที่สุด (Minimally Supervised) โดยแนวคิดพื้นฐานของวิธีการในกลุ่มนี้ จะสัมพันธ์กับหลักการ Bootstrapping ซึ่งใช้ประโยชน์จากลักษณะหรือรูปแบบการเกิดที่ซ้ำกันร่วมกับการใช้รายการชื่อเริ่มต้นขนาดเล็ก (Seed Name List) กฎเริ่มต้น (Seed Rule) หรือ ข้อมูลที่มีการกำกับขนาดเล็ก

แนวทางแบบผสม

วิธีการในกลุ่มนี้ เป็นวิธีการที่ผสมกันระหว่างการสร้างกฎโดยผู้เชี่ยวชาญ และการใช้หลักการทางสถิติ หรือหลักการฝึกฝนให้ระบบเรียนรู้ (Machine Learning) ตัวอย่างของระบบในกลุ่มนี้ ได้แก่ ระบบ LTG (Mikheev, Grover and Moens 1998 : 1-11) ซึ่งใช้กฎร่วมกับเทคนิคการเปรียบเทียบส่วนคำด้วยการใช้สถิติ (Statistical Partial Matching Technique) ระบบนี้ มีการทำงานเป็นลำดับขั้นตอนที่สอดแทรกสลับกัน โดยแต่ละขั้นตอน จะใช้ข้อสนเทศจากผลลัพธ์การสกัด NE ของขั้นตอนก่อนหน้าในการเปรียบเทียบกับคำหรือบางส่วนของคำ

การนำความรู้ภายนอกมาใช้ในการสกัด NE และความสัมพันธ์

งานวิจัยในการสกัด NE และความสัมพันธ์ทั่วไปนั้น มักจะมีการใช้ข้อมูลความรู้จากภายนอก โดยมีวัตถุประสงค์เพื่อให้เป็นข้อสนเทศเพื่อช่วยในการตัดสินใจ นอกจากนี้ ในงานวิจัยบางงาน ยังใช้ข้อมูลความรู้จากภายนอกเพื่อชดเชยการมีปริมาณคลังเอกสารเพื่อฝึกฝนระบบไม่เพียงพอ ดังเช่น งานของ Stevenson และ Gaizauskas (Stevenson and Gaizauskas 2000 : 290-295) ใช้ข้อมูลจากภายนอกมาช่วยในการสกัด NE ซึ่งในรายชื่อนั้นประกอบด้วย ชื่อองค์กร สถานที่ ชื่อคน คำนำหน้าชื่อต่างๆ

อย่างไรก็ตามการใช้คลังชื่อเพียงอย่างเดียวในการสกัด NE โดยใช้หลักการเปรียบเทียบคำกับคลังชื่อ มักก่อให้เกิดปัญหาหลายประการ ข้อแรกคือ NE สามารถเกิดขึ้นใหม่ได้ตลอดเวลา ทำให้การเก็บชื่อทั้งหมดไว้ในคลังชื่อ ไม่สามารถทำได้โดยง่าย ทั้งยังอาจก่อให้เกิดปัญหาในการควบคุมขนาดคลังชื่อด้วย นอกจากนี้ยังมีปัญหาในกรณีที่มีคำพ้องรูป แต่มีความหมายต่างกัน ซึ่งทำให้ระบบไม่สามารถตัดสินประเภทของชื่อนั้นได้ หรือในกรณีที่ NE มีรูปเดียวกับคำทั่วไป ซึ่งอาจทำให้ระบบเกิดความสับสน และสกัดคำทั่วไปเป็น NE อย่างไรก็ดี คลังชื่อหรือพจนานุกรมชื่อ ก็เป็นฐานความรู้ที่สำคัญที่ระบบสกัด NE เกือบทุกระบบนำไปใช้ เพื่อช่วยในการวิเคราะห์ โดยพจนานุกรมชื่อเหล่านี้ อาจสร้างขึ้นโดยใช้คนเป็นผู้รวบรวม หรือสร้างขึ้นมาจากคลังเอกสาร อย่างไรก็ดี Mark Stevenson และ Robert Gaizauskas ได้แสดงให้เห็นว่าภายใต้ข้อกำหนดเดียวกัน ระบบที่ใช้พจนานุกรมที่สร้างจากคลังเอกสาร (corpus-derived list) มีประสิทธิภาพสูงกว่าระบบที่ใช้พจนานุกรมที่สร้างโดยคน (hand-crafted list) และนอกจากนี้ การรวมพจนานุกรมที่สร้างโดยคนเข้ากับพจนานุกรมที่สร้างจากคลังเอกสาร ก็มักจะช่วยปรับปรุงประสิทธิภาพของระบบให้สูงขึ้น

ปัญหาของการสกัด NE และความสัมพันธ์ ในภาษาไทย (Chanlekha and Kawtrakul 2004)

ในการสกัด NE โดยทั่วไปสามารถแบ่งได้เป็น 2 ส่วนหลักคือ

1. การหาหาตำแหน่งและขอบเขตของ NE (Word Boundary Problem) ปัญหาในการหาตำแหน่งและขอบเขตของ NE เกิดจากภาษาไทยไม่มีข้อมูลจากลักษณะตัวอักษร ส่งผลให้การหาตำแหน่ง NE มีปัญหามากขึ้นโดยเฉพาะเมื่อ NE มีลักษณะเหมือนคำทั่วไป มีลักษณะเหมือนคำ

ทั่วไป เช่น คำกริยา (v) คำนาม (n) หรือนามวลี (NP) ดังตัวอย่าง สุภาพ(NE)มีกิริยามารยาท*สุภาพ* (v)เรียบร้อย จะเห็นว่ามีคำที่เขียนเหมือนกัน(สุภาพ) แต่ทำหน้าที่ต่างกัน หรือกรณีที่ NE เกิดขึ้นจากการรวมกันของคำทั่วไป หรือมีลักษณะแบบนามวลี การขาดข้อมูลจากลักษณะตัวอักษร ยังทำให้เกิดปัญหาในการระบุขอบเขต ตัวอย่างเช่น องค์การอาหารและเกษตรแห่งสหประชาชาติ เมื่อเป็นภาษาอังกฤษจะใช้ชื่อว่า Food and Agriculture Organization of United Nations ซึ่งจะเห็นว่าในภาษาอังกฤษนั้นมีการใช้ตัวอักษรพิมพ์ใหญ่ซึ่งทำให้สังเกตได้ง่ายว่าเป็นชื่อของหน่วยงาน ในขณะที่ในภาษาไทยไม่มีข้อมูลจากตัวอักษรเพื่อช่วยหาขอบเขต

2. การระบุประเภทของ NE (Category Problem) เกิดขึ้นเมื่อบริบทรอบ NE ไม่สามารถช่วยในการระบุประเภทของ NE นั้นได้ นอกจากนั้น NE ต่างประเภทกันยังสามารถอยู่ในบริบทที่คล้ายกันได้ เช่น การบินไทยเปิดเผยว่า และ อภิสิทธิ์เปิดเผยว่า

สำหรับการสกัด NE ภาษาไทยต้องประสบปัญหาหลายประการเนื่องจากคุณลักษณะของภาษาไทย ได้แก่ (1) ไม่มีข้อมูลจากลักษณะอักษรที่ช่วยในการหาตำแหน่ง รวมทั้งขอบเขตของ NE เช่น การใช้ตัวอักษรพิมพ์ใหญ่ในภาษาอังกฤษ (2) ไม่มีการใช้ช่องว่างเพื่อบ่งบอกขอบเขตของคำ ซึ่งทำให้เกิดปัญหาในการหาขอบเขตของ NE คุณลักษณะเหล่านี้ส่งผลให้เกิดปัญหา โดยเฉพาะอย่างยิ่งเมื่อNE มีลักษณะเหมือนนามวลี (Noun Phrase - NP) หรือประกอบด้วยคำหลายคำเรียงกัน

สำหรับการสกัดความสัมพันธ์ในภาษาไทยนั้นจะใช้วิธีการวิเคราะห์บริบทในประโยค ซึ่งปัญหาในการวิเคราะห์ประโยคนั้นมีลักษณะคล้ายกับการวิเคราะห์ประโยคในการสกัด NE ดังที่กล่าวมาแล้ว ดังนั้นจึงไม่ขอกล่าวซ้ำอีก

บทที่ 4

วิธีดำเนินการวิจัย

งานวิจัยนี้ต้องการสร้างระบบที่สามารถสกัด NE ทั้งหมด 4 ชนิด ได้แก่ ORG, PER, LOC, PRO จากนั้นจึงหาความสัมพันธ์ที่อยู่ระหว่าง NE ทั้งหมด 3 ชนิด ได้แก่ go_to, located_in, create มีขั้นตอนและวิธีดำเนินการดังนี้

ข้อมูลที่ใช้ในการวิจัย

วิทยานิพนธ์นี้ใช้คลังข้อมูลในการช่วยฝึกฝนระบบ ซึ่งคลังข้อมูลที่ใช้คือ คลังข้อมูล Orchid ของศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ* ซึ่งเป็นคลังข้อความด้านวิทยาศาสตร์และเทคโนโลยีภาษาไทยขนาดใหญ่ (ประมาณ 400,000 คำ) ที่มีการกำกับหน้าที่คำ (POS) ไว้แล้ว ที่สำคัญคือเป็นคลังข้อมูลภาษาไทยเดียว (ในขณะนี้) ที่อนุญาตให้นำไปใช้โดยไม่ต้องเสียค่าใช้จ่าย หรือเงื่อนไขใดๆ ส่วนข้อมูลที่นำมาใช้ในการทดสอบระบบนั้นจะใช้ข้อมูลข่าวหรือบทความที่เกี่ยวข้องกับด้านวิทยาศาสตร์และเทคโนโลยีจำนวน 200 ชุด

อุปกรณ์ที่ใช้ในการทดลอง

- เครื่องคอมพิวเตอร์แบบตั้งโต๊ะ 1 เครื่อง : CPU AMD Athlon 64 2800+ (1.8 GHz) หน่วยความจำขนาด 1.28 GB, Hard disk 80 GB
- เครื่องคอมพิวเตอร์โน้ตบุ๊ก 1 เครื่อง : CPU AMD Turion 64 ML-30 1.6 GHz หน่วยความจำขนาด 1 GB, Hard disk 60 GB
- ระบบปฏิบัติการ Windows XP

---

* สามารถดาวน์โหลดได้ที่ http://www.links.nectec.or.th/orchid/

- คลังข้อมูล Orchid
- ชุดข้อมูลข่าวและบทความต่างๆ ด้านวิทยาศาสตร์และเทคโนโลยี
- โปรแกรมตัดคำภาษาไทย KU Wordcut จาก NAiST-Lab มหาวิทยาลัยเกษตรศาสตร์ *
- ตัวแปลภาษา PHP

ขั้นตอนในการทดลอง

1. จัดเตรียมและวิเคราะห์ข้อมูล

1.1. การเตรียมข้อมูลสำหรับการสกัด NE

ข้อมูลที่จะนำมาใช้ในการสกัด NE นั้นจะมีลักษณะเป็นพจนานุกรมชื่อที่แบ่งประเภทแล้วซึ่งได้แก่ ORG, PER, LOC, PRO รวมไปถึงรายชื่อคำนำหน้าชื่อคนด้วย

สำหรับวิธีการจัดหารายชื่อต่างๆ นั้นแบ่งเป็นสองวิธีได้แก่ การแยกรายชื่อจากคลังข้อมูล Orchid และการหารายชื่อจากแหล่งข้อมูลอื่น

1.1.1. การแยกรายชื่อจากคลังข้อมูล Orchid

ข้อมูลในคลังข้อมูล Orchid นั้นมีลักษณะเป็นข้อความที่ถูกกำกับด้วยตัวอักษรที่ระบุหน้าที่ของข้อความนั้น (POS Tagged) (ดังภาพที่ 1) ซึ่งมีอยู่หลากหลายประเภท เช่น คำนาม คำกริยา คำบอกจำนวน คำวิเศษณ์ คำสรรพนาม ฯลฯ แต่ในที่นี้เราต้องการรายชื่อ NE ซึ่ง NE นั้นมีลักษณะเป็นคำนาม ซึ่งคำนามที่ต้องการแยกออกจากคลังข้อมูลคือประเภท นามเฉพาะซึ่งถูกกำกับด้วย NPRP (Proper noun), นามทั่วไปซึ่งกำกับด้วย NCMN (Common noun) และคำนำหน้าชื่อซึ่งกำกับด้วย NTTL (Title noun)

---

* ดาวน์โหลดได้ที่ http://naist.cpe.ku.ac.th/pkg/kucut-1.2.2_python25_fix.zip

```
ฯพณฯ รัฐมนตรีว่าการกระทรวงวิทยาศาสตร์ เทคโนโลยีและการพลังงาน//
ฯพณฯ/NTTL
<space>/PUNC
รัฐมนตรีว่าการ/NCMN
กระทรวงวิทยาศาสตร์ เทคโนโลยีและการพลังงาน/NPRP
//
#P3
#1
ประเทศไทยได้มีการปรับเปลี่ยนโครงสร้างในการพัฒนาเศรษฐกิจของประเทศ \\
จากประเทศเกษตรกรรมไปสู่ความเป็นประเทศอุตสาหกรรมมากยิ่งขึ้น//
ประเทศไทย/NPRP
ได้/XVAM
มี/VSTA
การ/FIXN
ปรับเปลี่ยน/VACT
โครงสร้าง/NCMN
ใน/RPRE
การ/FIXN
พัฒนา/VACT
เศรษฐกิจ/NCMN
ของ/RPRE
ประเทศ/NCMN
<space>/PUNC
จาก/RPRE
ประเทศ/NCMN
เกษตรกรรม/NCMN
ไปสู่/RPRE
ความ/FIXN
เป็น/VSTA
ประเทศอุตสาหกรรม/NCMN
มาก/ADVN
ยิ่งขึ้น/ADVN
//
```

ภาพที่ 1 ตัวอย่างข้อมูลในคลังข้อมูล Orchid

เมื่อได้ทำการแยกข้อมูลออกจากคลังข้อมูลเรียบร้อยแล้ว รายชื่อที่ได้มานั้นประกอบด้วย คำนำหน้าชื่อจำนวน 44 รายการ, คำนามทั่วไป 5,905 รายการ, คำนามเฉพาะ 1,432 รายการ จากนั้นนำคำนามเฉพาะและคำนามทั่วไปมารวมกันและตัดรายชื่อที่ซ้ำกันออกไป ทำให้ได้รายชื่อคำนามทั้งสิ้น 7,328 รายการ จากนั้นนำคำนามที่รวมกันแล้วนี้มาทำการวิเคราะห์โดยการคัดเลือก(ด้วยมือ)ทีละคำว่าคำใดสมควรจัดอยู่ใน NE ประเภทใด หรือคำใดไม่ใช่ NE ให้ตัดทิ้งไป เพื่อทำการคัดแยกประเภทให้ออกมาเป็นชนิด ORG, PER, LOC, PRO ซึ่งเมื่อผ่านการแยกแล้ว ทำให้ได้รายชื่อประเภทต่างๆ ดังนี้ ORG จำนวน 690 รายการ, PER จำนวน 1,070 รายการ LOC จำนวน 356 รายการ, PRO จำนวน 5,415 รายการ

1.1.2. การหารายชื่อจากแหล่งข้อมูลอื่น

เนื่องจากรายชื่อที่ได้จากคลังข้อมูล Orchid นั้นยังมีปริมาณที่น้อยเกินไป ประกอบกับคลังข้อมูล Orchid นั้นถูกสร้างมานานแล้ว (ปี พ.ศ. 2540 (Nook 2551)) ทำให้ข้อมูลที่ได้มาจากคลังข้อมูลนี้จึงไม่มีความทันสมัย ดังนั้นจึงต้องมีการหารายชื่อจากแหล่งข้อมูลอื่นๆ เข้ามาเพิ่มเติม โดยแบ่งตามประเภทของ NE ดังนี้

1.1.2.1. ORG

สำหรับประเภท ORG ซึ่งเป็นประเภทองค์กร ได้มีการนำข้อมูลชื่อองค์กรต่างๆ มาเพิ่มเติมดังนี้

- รายชื่อผู้ผลิตซอฟต์แวร์ป้องกันไวรัส (Microsoft 2008) จำนวน 24 รายการ
- รายชื่อสถาบันการเงิน (ธนาคารแห่งประเทศไทย 2551) จำนวน 117 รายการ
- รายชื่อบริษัทจดทะเบียนในตลาดหลักทรัพย์ (ตลาดหลักทรัพย์แห่งประเทศไทย 2551) จำนวน 549 รายการ
- รายชื่อผู้ผลิตฮาร์ดแวร์คอมพิวเตอร์ จากเว็บไซต์ Wikipedia (http://en.wikipedia.org/wiki/List_of_computer_hardware_manufacturers)* จำนวน 264 รายการ
- รายชื่อยี่ห้อสินค้า (ที่เป็นชื่อเดียวกับชื่อบริษัท) (Yopi 2008) จำนวน 255 รายการ
- รายชื่อหน่วยงานราชการและรัฐวิสาหกิจ (กระทรวงพลังงาน, สำนักงานนโยบายและพลังงาน 2551) จำนวน 945 รายการ

จากนั้นนำรายชื่อทั้งหมดมารวมกันและตัดรายชื่อที่ซ้ำกันออก จะได้รายชื่อสำหรับประเภท ORG จำนวน 2,062 รายการ

---

* เข้าถึงเมื่อ 5 กันยายน 2551

1.1.2.2. PER

สำหรับประเภท PER ซึ่งเป็นประเภทที่แสดงถึงบุคคล ได้มีการนำชื่อบุคคลจากแฟ้มข้อมูล firstname และ lastname ของโปรแกรมตัดคำ KU Wordcut มาเพิ่มเติม ซึ่งมีจำนวนทั้งสิ้น 30,875 รายการ

1.1.2.3. LOC

สำหรับประเภท LOC ซึ่งเป็นประเภทที่แสดงถึงสถานที่ต่างๆ ได้มีการนำชื่อสถานที่ต่างๆ มาเพิ่มเติมดังนี้

- แฟ้มข้อมูล country ของโปรแกรมตัดคำ KU Wordcut จำนวน 206 รายการ
- รายชื่อประเทศ ดินแดน เขตการปกครอง และเมืองหลวง (ร่างประกาศสำนักนายกรัฐมนตรี และประกาศราชบัณฑิตยสถาน 2544) จำนวน 1,408 รายการ
- รายชื่อจังหวัด เขต อำเภอ และกิ่งอำเภอในประเทศไทย (ราชบัณฑิตยสถาน 2551 : 5-57) จำนวน 1,003 รายการ
- รายชื่อทะเลสำคัญในโลก (จำเรียง จันทรประภา, ผู้รวบรวม 2551) จำนวน 63 รายการ
- รายชื่อทางแยกในกรุงเทพมหานคร จากเว็บไซต์วิกิพีเดีย (http://th.wikipedia.org/wiki/รายชื่อทางแยกในกรุงเทพมหานคร)* จำนวน 130 รายการ
- รายชื่อถนนในกรุงเทพมหานคร จากเว็บไซต์วิกิพีเดีย (http://th.wikipedia.org/wiki/หมวดหมู่:ถนนในกรุงเทพมหานคร)** จำนวน 112 รายการ
- รายชื่อสะพานที่มีชื่อเสียงในประเทศไทย จากเว็บไซต์วิกิพีเดีย (http://th.wikipedia.org/wiki/สะพาน)*** จำนวน 27 รายการ

---

* เข้าถึงเมื่อ 20 สิงหาคม 2551

** เข้าถึงเมื่อ 20 สิงหาคม 2551

*** เข้าถึงเมื่อ 20 สิงหาคม 2551

- รายชื่อสถานที่สำคัญ 76 จังหวัด (บ้านจอมยุทธ 2551) จำนวน 1,827 รายการ

1.1.2.3. PRO

สำหรับประเภท PRO ซึ่งเป็นประเภทที่แสดงถึงสิ่งของหรือสินค้าต่างๆ ได้มีการนำรายชื่อมาเพิ่มเติมดังนี้

- รายชื่อซอฟต์แวร์ป้องกันไวรัส จากเว็บไซต์ Wikipedia (http://en.wikipedia.org/wiki/List_of_antivirus_software)* จำนวน 51 รายการ

- รายชื่อฟรีซอฟต์แวร์ (Winaddons.com 2007) จำนวน 288 รายการ

สุดท้ายคือการนำรายชื่อที่ได้จากคลังข้อมูล Orchid และข้อมูลจากแหล่งอื่นๆ มารวมกันเพื่อเตรียมนำไปใช้ในการสกัด NE ซึ่งรายชื่อทั้งหมดที่ได้มามีดังนี้ ORG จำนวน 2,695 รายการ, PER จำนวน 31,497 รายการ, LOC จำนวน 4,664 รายการ, และ PRO จำนวน 6,107 รายการ

1.2. จัดหาข้อความข่าวหรือบทความ

ข้อความข่าวหรือบทความนี้จะถูกนำไปใช้ในการทดสอบการสกัด NE และความสัมพันธ์ ซึ่งข้อความเหล่านี้จะมีเนื้อหาเกี่ยวกับเทคโนโลยีในวงการคอมพิวเตอร์ ในช่วงปี พ.ศ. 2546-2550 จากเว็บไซต์เออาร์ไอพี (Arip 2551) จำนวน 300 ข้อความ ซึ่งแต่ละข้อความมีความยาวไม่เกิน 4 บรรทัดสำหรับขนาดตัวอักษร 16 (Angsana) ในหน้ากระดาษขนาด A4

1.3. การสร้างกฎฮิวริสติกเพื่อใช้ในการสกัดความสัมพันธ์

กฎฮิวริสติกที่นำมาใช้ จะแบ่งเป็น 2 ประเภทคือ คำกริยา (Verb) และคำบุพบท (Preposition) โดยทั้งสองประเภทจะมีหลักการและโครงสร้างเหมือนกันทั้งหมด

---

* เข้าถึงเมื่อ 11 กันยายน 2551

รูปแบบของกฎนี้จะมีลักษณะเป็นแบบแผน (Pattern) เพื่อให้ระบบใช้ตรวจสอบโดยจะมีคำสำคัญที่เป็นกริยาหรือคำบุพบท ซึ่งแต่ละคำจะมีคุณสมบัติแตกต่างกัน คุณสมบัติเหล่านี้จะถูกใช้ในขณะที่มีการสกัดความสัมพันธ์ โดยเมื่อระบบพบคำกริยาหรือคำบุพบทในบริบทใดๆ คุณสมบัติเหล่านี้จะถูกนำมาประเมินด้วย เพื่อเป็นการยืนยันว่าบริบทนั้นๆ มีความสัมพันธ์ตามที่ได้พบคำสำคัญอยู่หรือไม่ และคุณสมบัติต่างๆ ที่นำมาใช้ในการประเมินความสัมพันธ์มีดังนี้ (ดูตัวอย่างได้ที่หน้า 90-95)

- **word** คือคำสำคัญหลักที่เป็นคำกริยาหรือคำบุพบทที่ระบบจะค้นหา
- **type** คือประเภทของความสัมพันธ์ ประกอบด้วย create, goto, located_in
- **ne1** คือประเภทของ NE ทางซ้ายมือที่ต้องการในบริบท หมายความว่า NE ทางด้านซ้ายมือจะต้องเป็นประเภทเดียวกับค่าที่ทำหนดไว้ หากไม่ใช่จะไม่ถือว่ามีความสัมพันธ์
- **unwant_ne1** คือประเภทของ NE ทางด้านซ้ายมือที่ไม่ต้องการของบริบท หมายความว่า NE ทางด้านซ้ายมือจะต้องไม่เป็นประเภทเดียวกับที่ถูกกำหนดไว้ หากเป็นประเภทเดียวกันจะไม่ถือว่ามีความสัมพันธ์
- **ne2** คือประเภทของ NE ทางด้านขวามือที่ต้องการในบริบท หมายความว่า NE ทางด้านขวามือจะต้องเป็นประเภทเดียวกับค่าที่กำหนดไว้ หากไม่ใช่จะไม่ถือว่ามีความสัมพันธ์
- **unwant_ne2** คือประเภทของ NE ทางด้านขวามือที่ไม่ต้องการของบริบท หมายความว่า NE ทางด้านขวามือจะต้องไม่เป็นประเภทเดียวกับที่ถูกกำหนดไว้ หากเป็นประเภทเดียวกันจะไม่ถือว่ามีความสัมพันธ์
- **unwant_pair_ne1 และ unwant_pair_ne2** คือคู่ของประเภท NE ที่ไม่ต้องการในด้านซ้ายและด้านขวา (unwant_pair_ne1 และ unwant_pair_ne2 ตามลำดับ) ในบริบท ยกตัวอย่างเช่น unwant_pair_ne1 มีค่า PRO และ unwant_pair_ne2 มีค่า ORG หมายความว่าในบริบทนั้นจะมี NE ทางด้านซ้ายจะมีค่าเป็น PRO และด้านขวามีค่าเป็น ORG อยู่ในบริบทเดียวกันไม่ได้
- **direction** คือทิศทางที่ NE จะมีความสัมพันธ์กัน โดยปกติจะไม่ถูกกำหนดค่าไว้ ซึ่งจะหมายความว่าเป็นความสัมพันธ์ทางด้านซ้ายมาขวาตามปกติ แต่หากมีการกำหนดค่าจะถูกกำหนดเป็น “left” หมายความว่า NE ในบริบทนั้นจะมีความสัมพันธ์จากขวามาซ้าย ยกตัวอย่างเช่น “|PRO|เมนบอร์ด|/PRO| ของ คอมพ์ รุ่นใหม่ จะ มี |PRO|ชิปเซต|/PRO|” จะเห็นว่าประโยคนี้มี NE

ได้แก่ เมนบอร์ด และ ชิปเซต และมีคำสำคัญที่บ่งบอกถึงความสัมพันธ์คือคำว่า “มี” ซึ่งแปลว่า ชิปเซตติดตั้งอยู่ในเมนบอร์ด (ความสัมพันธ์แบบ located_in) แต่จากตำแหน่งคำในประโยคจะเห็นว่าเป็นความสัมพันธ์จากขวามาซ้าย (คำว่าชิปเซตอยู่หลังจากคำว่าเมนบอร์ด) ดังนั้นคุณสมบัติ direction ในความสัมพันธ์นี้จะต้องถูกกำหนดเป็น “left”

- **unwant** คือคำที่ไม่ต้องการให้เกิดในบริบทที่พบคำสำคัญ หมายความว่าในบริบทที่พบคำสำคัญจะมีคำที่ตรงกับที่กำหนดไว้ใน unwant ไม่ได้ นอกจากนี้ในคุณสมบัตินี้ยังมีการตรวจสอบคำที่ไม่ต้องการได้หลายวิธี ดังตัวอย่างเช่น คำว่า “ลง” (ความสัมพันธ์ประเภท goto) มีค่า unwant คือ “แป.,.มือ” หมายความว่ามีคำที่ไม่ต้องการสองคำคือ “แป.” และ “.มือ” (แบ่งโดยเครื่องหมาย “,”) ซึ่งระบบจะค้นพบคำใดคำหนึ่งก็ได้ หากพบแล้วจะถือว่าบริบทนั้นไม่มีความสัมพันธ์ ส่วนเครื่องหมาย “.” นั้นหมายความว่าคำนั้นจะต้องอยู่ติดกับคำสำคัญ เช่น “แป.” มีจุดอยู่ด้านหลัง แปลว่า “แป” จะต้องอยู่ติดกับคำว่า “ลง” ทางด้านหน้า (แปลง) ส่วนคำว่า “มือ” มีจุดอยู่ด้านหน้า แปลว่า “มือ” จะต้องอยู่ติดกับคำว่า “ลง” ทางด้านหลัง (ลงมือ) แต่ถ้าหากไม่มีเครื่องหมายจุด แปลว่าคำนั้นจะอยู่ ณ ตำแหน่งใดก็ได้ในบริบท

- **want** คือคำที่ต้องการให้เกิดในบริบทที่พบคำสำคัญ หมายความว่าในบริบทที่พบคำสำคัญจะต้องพบคำที่ถูกกำหนดไว้ใน want มิเช่นนั้นจะถือว่าไม่มีความสัมพันธ์ ส่วนขั้นตอนในการตรวจสอบหาคำที่ต้องการนั้นจะใช้วิธีเดียวกับ unwant แต่จะมีคำว่า “prep” (preposition) เพิ่มเข้ามา ซึ่งจะมีเฉพาะคำสำคัญที่เป็นคำกริยาเท่านั้น หมายความว่าคำกริยาใดที่มี “prep” กำหนดเอาไว้ใน want นั้นจะต้องการคำบุพบท (ที่มีประเภทความสัมพันธ์เดียวกันกับคำกริยา) เพิ่มเข้ามาในบริบทนั้นด้วยจึงจะสามารถมีความสัมพันธ์เกิดขึ้นได้

- **want_after_ne2** มีหน้าที่เหมือนกับ want แต่คำที่อยู่ใน want_after_ne2 จะต้องเกิดขึ้นภายในบริบทถัดไป (หลัง NE ด้านขวาของบริบทปัจจุบัน) จึงจะเกิดความสัมพันธ์

- **adjacent** คือการบ่งบอกถึงตำแหน่งของคำสำคัญในบริบทนี้ว่าจะต้องอยู่ติดกับ NE ทางด้านขวาหรือไม่ ซึ่งคุณสมบัตินี้จะมีค่าเป็น 0 หรือ 1 ซึ่งถ้าหากเป็นค่า 0 หมายความว่าคำสำคัญจะอยู่ในตำแหน่งใดก็ได้ (ไม่จำเป็นต้องอยู่ติดกับ NE ด้านขวา) แต่หากเป็นค่า 1 หมายความว่าคำสำคัญจะต้องอยู่ติดกับ NE ทางด้านขวาในระยะห่างไม่เกินค่าที่กำหนดเอาไว้ในคุณสมบัติ len_to_ne2 ซึ่งจะกล่าวถึงในลำดับถัดไป

- **len_to_ne2** คือระยะห่างของคำสำคัญจาก NE ทางด้านขวาของบริบท ซึ่งระยะห่างนั้นสามารถวัดจากหน่วยคำที่ถูกแบ่งโดยโปรแกรมตัดคำ คุณสมบัตินี้จะถูกพิจารณาต่อเมื่อค่าใน adjacent เป็น 1 โดยค่าในคุณสมบัตินี้จะถูกกำหนดเป็นตัวเลข หมายถึงระยะห่างของคำสำคัญจาก NE ทางขวาจะต้องไม่เกินค่าตัวเลขที่กำหนดเอาไว้ ยกตัวอย่างเช่น "|ORG|Epson|/ORG| _ ผู้ **ผลิต** |PRO|เครื่อง พิมพ์ อิงค์เจ็ต|/PRO|" ("ผลิต" มีค่า len_to_ne2 เท่ากับ 1) หรือ "โอน |PRO|ข้อมูล|/PRO| ผ่าน การ เชื่อมต่อ ลักษณะ นี้ **ไป** ยัง |PRO|คอมพิวเตอร์|/PRO|" ("ไป" มีค่า len_to_ne2 เท่ากับ 2) เป็นต้น

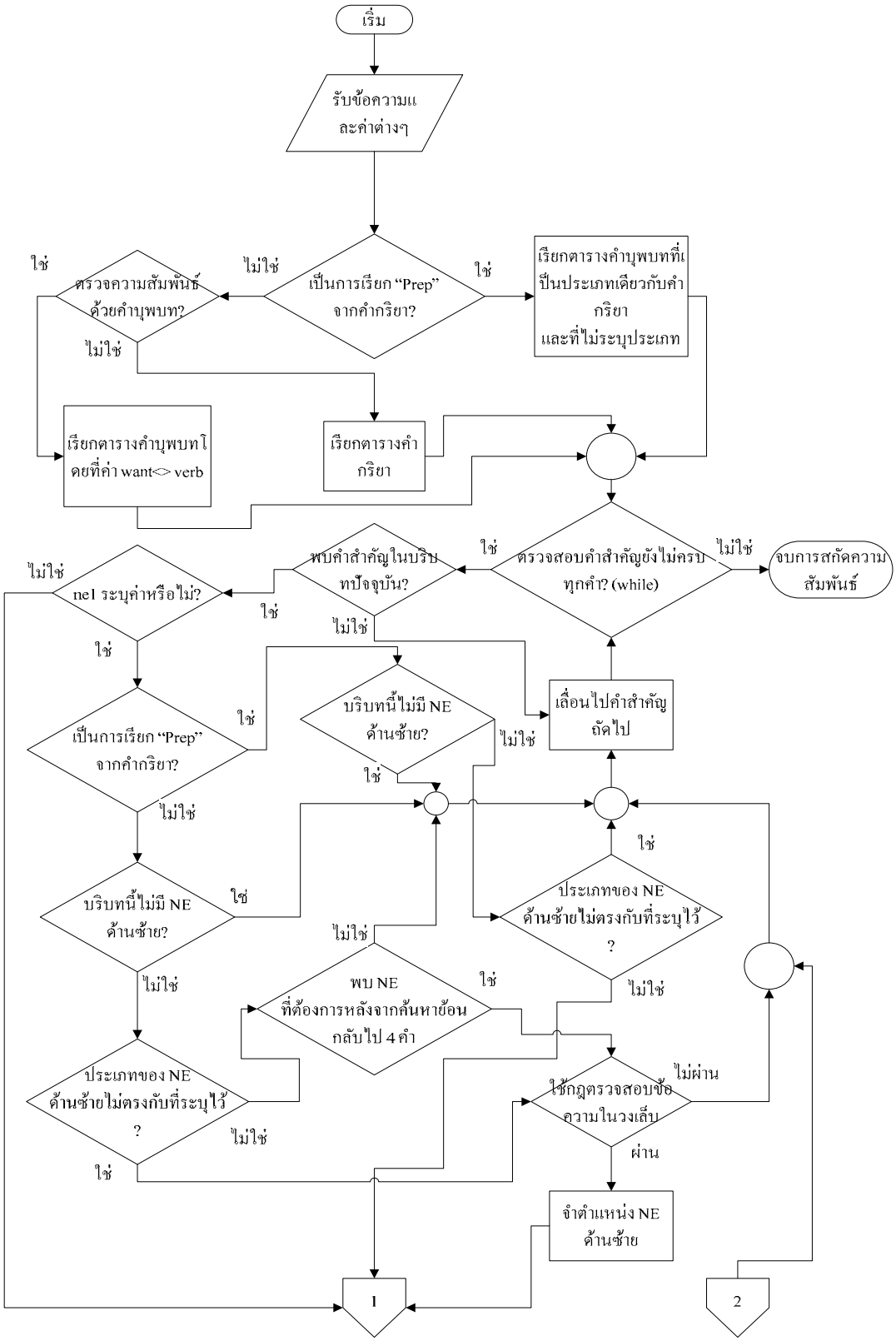
- **ncr_want (next context relation_want)** โดยปกติจะไม่มีค่าหรืออาจมีค่าเป็น 0 ซึ่งจะหมายความว่าระบบจะไม่พิจารณาคุณสมบัตินี้ แต่หากมีค่าเป็น 1 จะหมายความว่าคำสำคัญนั้นมีความสัมพันธ์เกิดขึ้นอยู่ในบริบทถัดจากบริบทปัจจุบัน ยกตัวอย่างเช่น "|PER|ผู้ ไม่ หวังดี|/PER| สามารถ **ส่ง** |PRO|โค้ด|/PRO| อันตราย เข้า ไป ทำงาน ใน |PRO|พีซี|/PRO|" จากตัวอย่างบริบทที่มีคำสำคัญคือบริบทที่มีคำว่า "ส่ง" แต่ความสัมพันธ์ที่เกิดขึ้นจริงนั้นเกิดขึ้นในบริบทถัดไป

- **may_want** มีหน้าที่เหมือนกับ want แต่จะถูกพิจารณาต่อเมื่อการพิจารณาในข้อ want, want_after_ne2, และ adjacent อย่างใดอย่างหนึ่งเป็นเท็จ

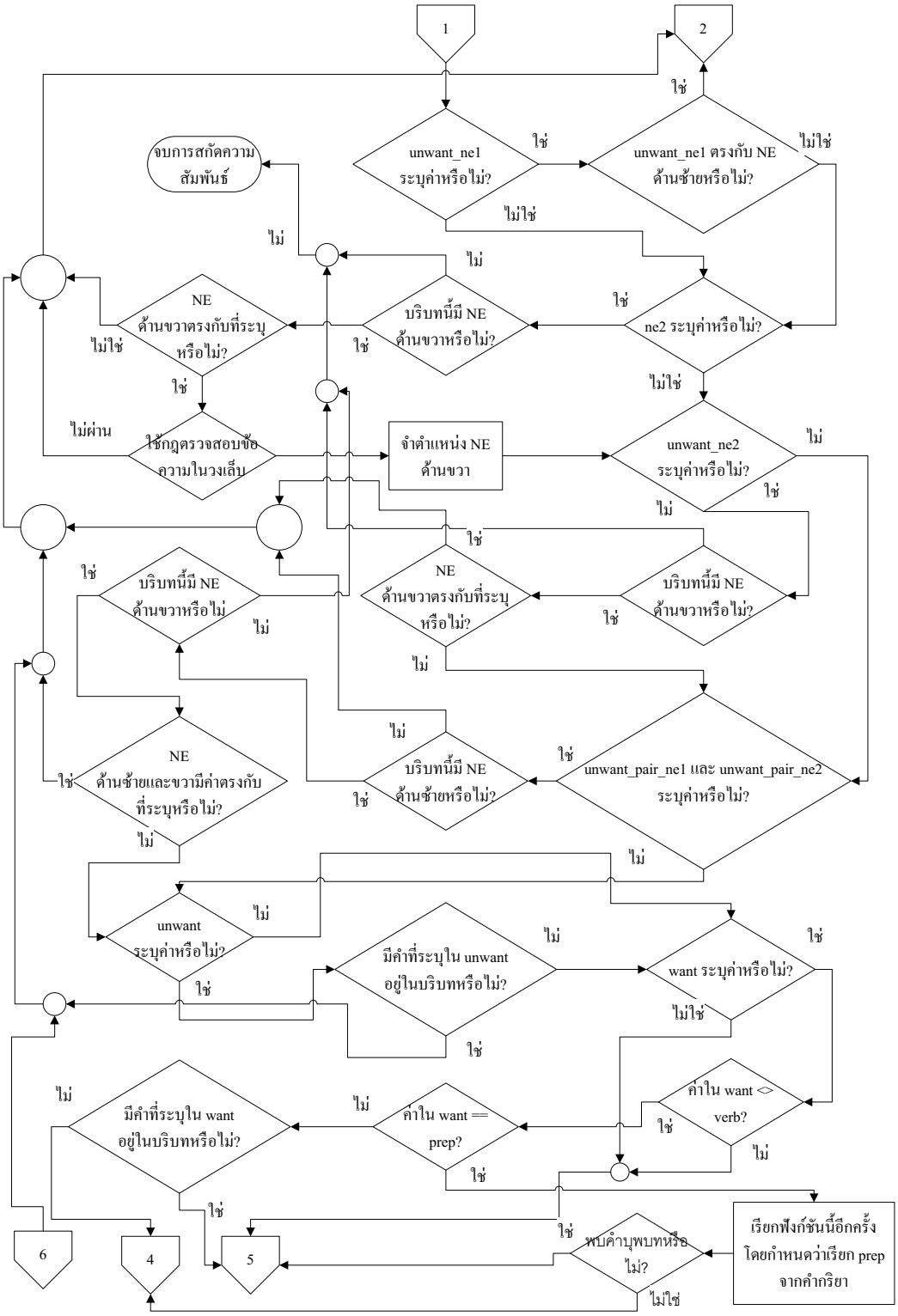
- **len_from_ne1** มีหน้าที่เหมือนกับ len_to_ne2 แต่จะเป็นการตรวจสอบระยะห่างจาก NE ทางด้านซ้ายแทน และไม่ได้ขึ้นกับคุณสมบัติ adjacent

1.3.1. ขั้นตอนการทำงานในส่วนของการสกัดความสัมพันธ์โดยใช้คำสำคัญ

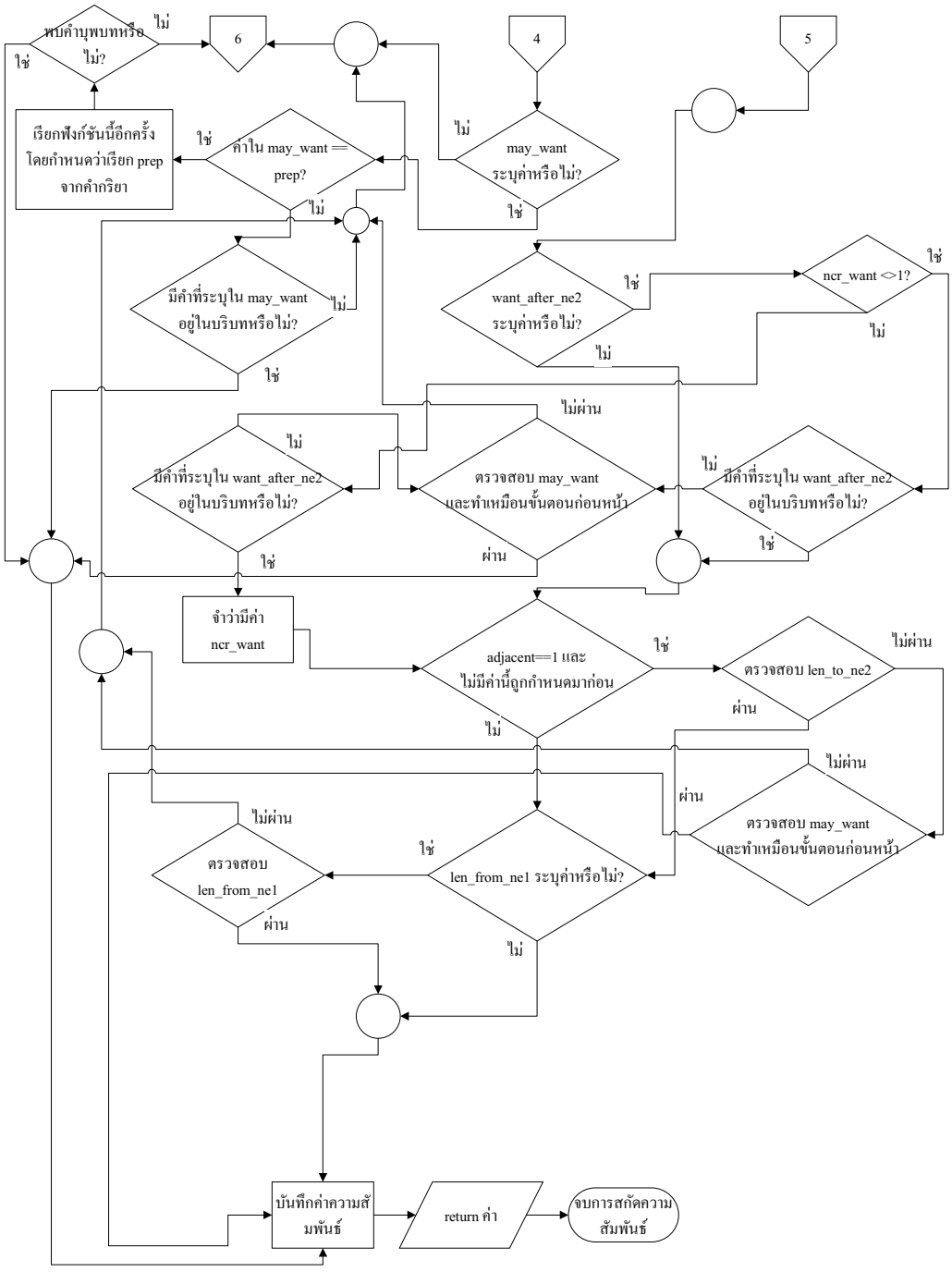
ในส่วนของการสกัดความสัมพันธ์โดยการใช้คำสำคัญนั้น มีขั้นตอนในการทำงานดังที่แสดงในภาพที่ 2



ภาพที่ 2 ผังแสดงการสกัดความสัมพันธ์ ในส่วนของการตรวจสอบคำสำคัญ


1
2
ใช่
unwant_ne1
ระบุค่าหรือไม่?
ใช่
unwant_ne1 ตรงกับ NE
ด้านซ้ายหรือไม่?
ไม่ใช่
จบการสกัดความ
สัมพันธ์
ไม่ใช่
ไม่
ไม่
ne2 ระบุค่าหรือไม่?
ใช่
บริบทนี้มี NE
ด้านขวาหรือไม่?
ใช่
NE
ด้านขวาตรงกับที่ระบุ
หรือไม่?
ไม่ใช่
ไม่ใช่
ใช่
ไม่ผ่าน
ใช้กฎตรวจสอบข้อ
ความในวงเล็บ
จำตำแหน่ง NE
ด้านขวา
unwant_ne2
ระบุค่าหรือไม่?
ไม่
ใช่
ไม่
ใช่
ใช่
บริบทนี้มี NE
ด้านขวาหรือไม่
ไม่
NE
ด้านขวาตรงกับที่ระบุ
หรือไม่?
ใช่
บริบทนี้มี NE
ด้านขวาหรือไม่?
ไม่
ไม่
ใช่
NE
ด้านซ้ายและขวามีค่าตรงกับ
ที่ระบุหรือไม่?
ใช่
บริบทนี้มี NE
ด้านซ้ายหรือไม่?
ใช่
unwant_pair_ne1 และ unwant_pair_ne2
ระบุค่าหรือไม่?
ไม่
ไม่
ไม่
unwant
ระบุค่าหรือไม่?
ไม่
ใช่
มีคำที่ระบุใน unwant
อยู่ในบริบทหรือไม่?
ไม่
want ระบุค่าหรือไม่?
ใช่
ไม่ใช่
ใช่
ไม่
มีคำที่ระบุใน want
อยู่ในบริบทหรือไม่?
ไม่
ค่าใน want ==
prep?
ค่าใน want <>
verb?
ใช่
ไม่
ใช่
ใช่
ใช่
6
4
5
พบคำบุพบทหรือ
ไม่?
เรียกฟังก์ชันนี้อีกครั้ง
โดยกำหนดว่าเรียก prep
จากคำกริยา
ไม่ใช่

ภาพที่ 2 (ต่อ)



ภาพที่ 2 (ต่อ)

จากภาพที่ 2 แสดงให้เห็นขั้นตอนการทำงานของวิธีการสกัดความสัมพันธ์โดยใช้คำสำคัญ โดยในส่วนนี้มีลักษณะการทำงานแบบฟังก์ชัน ซึ่งจะรับค่าบริบททั้งหมด, NE

ทั้งหมด, ตำแหน่งของบริบทที่ต้องการค้นหา, NE ทางซ้ายและขวาในบริบทที่ต้องการค้นหา, ประเภทคำสำคัญที่ต้องการเรียก

ในขั้นตอนแรกจะเป็นการตรวจสอบประเภทการเรียกฐานข้อมูลคำสำคัญ โดยจะแบ่งเป็น การเรียกคำกริยา, การเรียกคำบุพบท, และการเรียกคำบุพบทในขั้นตอนการตรวจสอบคำกริยา (จะถูกเรียกต่อเมื่อพบคำว่า “prep” ในคุณสมบัติของคำสำคัญ) จากนั้นจะเข้าสู่การทำงานแบบวนซ้ำ (while) โดยกำหนดเงื่อนไขให้ทำงานต่อเมื่อตรวจสอบคำสำคัญจากฐานข้อมูลยังไม่ครบทุกคำ และในทุกๆ รอบจะเปลี่ยนคำสำคัญไปเรื่อยๆ จนกว่าจะครบ

เมื่อเข้ามาสู่การทำงานแบบวนซ้ำแล้วขั้นตอนแรกจะทำการตรวจสอบคำสำคัญในรอบปัจจุบันว่าปรากฏอยู่ในบริบทที่กำลังตรวจสอบอยู่หรือไม่ หากไม่ใช่ให้ไปเริ่มรอบการทำงานต่อไป (continue) แต่หากใช่จะเริ่มทำการตรวจสอบคุณสมบัติในคำสำคัญปัจจุบัน โดยเริ่มที่ ne1 โดยจะตรวจสอบว่าในคุณสมบัติ ne1 ของคำสำคัญปัจจุบันได้ถูกกำหนดค่าไว้หรือไม่ ซึ่งหากไม่ใช่ ให้ทำการตรวจสอบคุณสมบัติต่อไป แต่หากใช่จะทำการตรวจสอบว่าในบริบทนี้มี NE ด้านซ้ายอยู่หรือไม่ หากไม่ใช่ให้ไปเริ่มรอบการทำงานต่อไป แต่หากใช่ให้ตรวจสอบว่าชนิด NE ด้านซ้ายของบริบทนี้ตรงกับที่คำสำคัญต้องการหรือไม่ หากใช่ให้จำตำแหน่งของ NE นั้นไว้เป็น NE ด้านซ้ายหากมีความสัมพันธ์เกิดขึ้น หากไม่ใช่ให้ทำการค้นหาชนิดของ NE ย้อนกลับไปยังต้นข้อความในระยะ NE 4 คำ (แต่หากเป็นการตรวจสอบคำสำคัญประเภทคำบุพบทซึ่งเป็นการเรียกจากค่า “prep” จากคำกริยา จะไม่ต้องทำในขั้นตอนนี้) โดยหากพบ NE ที่ต้องการในระยะ 4 คำ ให้จำตำแหน่งของ NE นั้นไว้เป็น NE ด้านซ้ายหากมีความสัมพันธ์เกิดขึ้น แต่หากไม่พบให้ไปเริ่มรอบการทำงานต่อไป โดยในการจำตำแหน่งของ NE นั้นจะมีการตรวจสอบด้วยว่าบริบทที่ทำการตรวจสอบนั้นอยู่ในวงเล็บหรือไม่ได้อยู่ในวงเล็บ โดยหากอยู่ในวงเล็บ NE ที่จะนำมามีความสัมพันธ์นั้นจะต้องอยู่ในวงเล็บ แต่หากไม่ได้อยู่ในวงเล็บ NE ก็จะต้องไม่ได้อยู่ในวงเล็บเช่นเดียวกัน โดยหากไม่ตรงตามเงื่อนไขนี้ให้ไปเริ่มรอบการทำงานต่อไป ขั้นตอนต่อไปคือการตรวจสอบคุณสมบัติ unwant_ne1 โดยจะตรวจสอบว่ามีการระบุค่าเอาไว้หรือไม่ หากไม่ได้ระบุค่าเอาไว้ให้ข้ามไปตรวจสอบคุณสมบัติต่อไป แต่หากถูกระบุค่าเอาไว้ให้ตรวจสอบว่า NE ด้านซ้ายในบริบทนี้ตรงกับที่ระบุเอาไว้ในคุณสมบัตินี้หรือไม่ หากใช่ให้ไปเริ่มรอบการทำงานต่อไป แต่หากไม่ใช่ให้ไปตรวจสอบคุณสมบัติ ne2 โดยตรวจสอบว่ามีการระบุค่าเอาไว้หรือไม่ หากไม่ใช่ให้ข้าม

ไปตรวจสอบคุณสมบัติต่อไป หากใช่ให้ตรวจสอบว่าบริบทที่ตรวจสอบนั้นมี NE ด้านขวาหรือไม่ หากไม่ใช่ให้ออกจากการตรวจสอบ (break) หากใช่ให้ตรวจสอบชนิดของ NE ที่กำหนดนั้นตรงกับชนิดของ NE ด้านขวาในบริบทหรือไม่ หากไม่ใช่ให้ไปเริ่มรอบการทำงานต่อไป หากใช่ให้ตรวจสอบวงเล็บเช่นเดียวกับในคุณสมบัติ ne1 หากตรวจสอบไม่ผ่านให้ไปเริ่มรอบการทำงานต่อไป หากผ่านให้จำตำแหน่งของ NE เป็น NE ด้านขวาหากมีความสัมพันธ์เกิดขึ้น ต่อไปคือการตรวจสอบว่า unwant_ne2 นั้นถูกกำหนดค่าไว้หรือไม่ หากไม่ใช่ให้ข้ามไปตรวจสอบคุณสมบัติต่อไป หากใช่ให้ตรวจสอบสอบว่าบริบทนี้มี NE ด้านขวาหรือไม่ หากไม่ใช่ให้ออกจากการตรวจสอบ หากใช่ให้ตรวจสอบว่า NE ด้านขวาตรงกับที่ระบุเอาไว้ใน unwant_ne2 หรือไม่ หากใช่ให้เริ่มรอบการทำงานใหม่ หากใช่ให้ตรวจสอบว่า unwant_pair_ne1 และ unwant_pair_ne2 (ทั้งคู่) ถูกระบุค่าเอาไว้หรือไม่ หากไม่ใช่ให้ข้ามไปตรวจสอบคุณสมบัติถัดไป หากใช่ให้ตรวจสอบว่ามี NE ด้านซ้ายหรือไม่ หากไม่ใช่ให้ไปเริ่มรอบการทำงานต่อไป หากใช่ให้ตรวจสอบว่ามี NE ด้านขวาหรือไม่ หากไม่ใช่ให้ออกจากการตรวจสอบ หากใช่ให้ตรวจสอบว่า NE ด้านซ้ายและด้านขวาตรงกับที่ระบุเอาไว้หรือไม่ หากใช่ให้ไปเริ่มรอบการทำงานต่อไป หากไม่ใช่ให้ตรวจสอบว่า unwant มีการระบุค่าเอาไว้หรือไม่ หากไม่ใช่ให้ข้ามไปตรวจสอบคุณสมบัติถัดไป หากใช่ให้ตรวจสอบว่ามีคำที่ระบุใน unwant อยู่ในบริบทหรือไม่ หากใช่ให้ไปเริ่มรอบการทำงานต่อไป หากไม่ใช่ให้ตรวจสอบว่า want ถูกระบุค่าไว้หรือไม่ หากไม่ใช่ให้ข้ามไปตรวจสอบคุณสมบัติถัดไป หากใช่ให้ตรวจสอบว่าค่าใน want นั้นไม่เท่ากับ “verb” ใช่หรือไม่ หากไม่ใช่ให้ข้ามไปตรวจสอบคุณสมบัติถัดไป หากใช่ให้ตรวจสอบว่าค่าใน want นั้นเท่ากับ “prep” หรือไม่ หากใช่ให้ทำการเรียกฟังก์ชันนี้อีกครั้ง โดยกำหนดให้เป็นการเรียกคำบุพบทจากคำกริยาและตรวจสอบว่าพบคำบุพบทในบริบทหรือไม่ หากใช่ให้ข้ามไปตรวจสอบคุณสมบัติถัดไป หากไม่ใช่ให้ไปตรวจสอบคุณสมบัติ may_want หากค่าใน want ไม่เท่ากับ “prep” ให้ตรวจสอบว่ามีคำที่ระบุใน want อยู่ในบริบทหรือไม่ หากใช่ให้ข้ามไปตรวจสอบคุณสมบัติถัดไป หากไม่ใช่ให้ตรวจสอบคุณสมบัติ may_want โดยตรวจสอบว่ามีถูกระบุค่าไว้หรือไม่ หากไม่ใช่ให้เริ่มการทำงานในรอบถัดไป หากใช่ให้ตรวจสอบว่าค่าใน may_want เท่ากับ “prep” หรือไม่ หากใช่ให้ทำเช่นเดียวกับขั้นตอนก่อนหน้า โดยหากไม่พบคำบุพบทให้เริ่มการทำงานในรอบถัดไป แต่หากพบคำบุพบทให้ออกจากการตรวจสอบความสัมพันธ์และบันทึกค่ารายละเอียดความสัมพันธ์ เช่น มีความสัมพันธ์เกิดขึ้นกับคำ

สำคัญปัจจุบันและตำแหน่ง NE ด้านซ้ายและขวา เป็นต้น แต่หากค่าใน may_want ไม่เท่ากับ “prep” ให้ตรวจสอบว่ามีค่าที่ระบุใน may_want อยู่ในบริบทหรือไม่ หากไม่พบค่าดังกล่าวให้เริ่มการทำงานในรอบถัดไป แต่หากพบให้ออกจากการตรวจสอบความสัมพันธ์และบันทึกค่ารายละเอียดความสัมพันธ์ แต่หากระบบไม่ได้ตรวจสอบคุณสมบัติ may_want และไม่ได้ถูกสั่งให้เริ่มการทำงานในรอบถัดไป ระบบจะตรวจสอบคุณสมบัติถัดไป คือ want_after_ne2 โดยทำการตรวจสอบว่ามีค่าระบุไว้หรือไม่ หากไม่พบให้ข้ามไปตรวจสอบคุณสมบัติถัดไป หากพบจะตรวจสอบว่า ncr_want ไม่เท่ากับ 1 ใช่หรือไม่ หากใช่ให้ตรวจสอบว่าค่าที่ระบุมาใน want_after_ne2 นั้นอยู่ในบริบทถัดไปหรือไม่ หากพบให้ข้ามไปตรวจสอบคุณสมบัติถัดไป หากไม่พบให้ไปตรวจสอบคุณสมบัติ may_want โดยทำเช่นเดียวกับขั้นตอนที่ได้กล่าวมาแล้ว แต่หาก ncr_want เท่ากับ 1 จะตรวจสอบว่ามีคำที่ระบุใน want_after_ne2 อยู่ในบริบทถัดไปหรือไม่ หากไม่พบให้ตรวจสอบคุณสมบัติ may_want แต่หากพบให้จำไว้ว่า ncr_want นั้นถูกกำหนดเป็น 1 เพื่อที่จะได้ทราบว่าความสัมพันธ์นั้นเกิดขึ้นในบริบทถัดไป(หากมีความสัมพันธ์เกิดขึ้น) จากนั้นจะตรวจสอบคุณสมบัติถัดไปคือ adjacent โดยจะตรวจสอบว่าค่าใน adjacent เท่ากับ 1 และไม่มีค่านี้ถูกกำหนดมาก่อนใช่หรือไม่ (ในส่วนนี้หมายความว่าอาจจะมีค่านี้ถูกกำหนดมาก่อนหน้า ซึ่งอาจเกิดขึ้นเมื่อคำกริยาเรียกตรวจสอบคำบุพบท (prep) ซึ่งคุณสมบัติของคำบุพบทนั้นอาจจะกำหนดให้ต้องอยู่ใกล้กับ NE ด้านขวาอยู่ก่อนแล้วก็เป็นได้ ดังนั้นเมื่อกลับมาตรวจสอบคำกริยาจึงไม่ต้องตรวจสอบค่านี้อีก เพราะในการเขียนข้อความหากมีคำบุพบทเกิดร่วมกันคำกริยาคำบุพบทนั้นจะเกิดขึ้นหลังจากคำกริยาเสมอ) ซึ่งหากเงื่อนไขเป็นเท็จจะข้ามไปตรวจสอบคุณสมบัติถัดไป แต่หากเป็นจริงจะตรวจสอบคุณสมบัติ len_to_ne2 ว่ามีระยะห่างจาก NE ด้านขวาตามที่กำหนดไว้หรือไม่ หากไม่ใช่ให้ไปตรวจสอบ may_want แต่หากใช่ให้ตรวจสอบคุณสมบัติ len_from_ne1 ว่ามีการระบุค่าเอาไว้หรือไม่ หากไม่ใช่จบการตรวจสอบความสัมพันธ์และบันทึกค่ารายละเอียดความสัมพันธ์ หากใช่ให้ตรวจสอบระยะห่างของคำสำคัญกับ NE ด้านซ้าย ว่าอยู่ในระยะที่กำหนดไว้หรือไม่ หากไม่ใช่ให้เริ่มรอบการทำงานใหม่ หากใช่ให้จบการตรวจสอบความสัมพันธ์และบันทึกค่ารายละเอียดความสัมพันธ์ แต่หากหลังจากทำงานจนครบรอบการทำงานทั้งหมดแล้ว (ตรวจสอบคำสำคัญทั้งหมดแล้ว) จนออกจากการทำงานวนซ้ำ ให้ถือว่าไม่มีความสัมพันธ์เกิดขึ้น

ทั้งนี้ในการจัดเตรียมคำสำคัญเพื่อนำในมาใช้ในการสกัดความสัมพันธ์นั้น ได้ทำการค้นหาจากการวิเคราะห์ข้อความต่างๆ ที่มีความสัมพันธ์ประเภทที่ต้องการค้นหาในวิทยานิพนธ์นี้ จนได้คำสำคัญประเภทคำกริยาจำนวน 48 คำ (แบ่งเป็นประเภท create 22 คำ, goto 10 คำ, และ located_in 16 คำ) และคำบุพบทจำนวน 22 คำ (แบ่งเป็นประเภท create 5 คำ, goto 6 คำ, located_in 10 คำ, และคำที่สามารถใช้ได้กับทุกประเภท 1 คำ) สำหรับคำบุพบทนั้นมีคำที่ลักษณะพิเศษอยู่หนึ่งคำ คือคำว่า “ที่” ซึ่งเป็นคำที่ไม่เจาะจงว่าเป็นประเภทใด แต่สามารถเติมเต็มประโยคต่างๆ ให้สมบูรณ์ได้ ยกตัวอย่างเช่น ไปที่ (goto), อยู่ที่ (located_in) เป็นต้น ดังนี้จึงมีความจำเป็นที่จะต้องรวมคำนี้เอาไว้ด้วย

2. นำข้อความที่จัดเตรียมไว้ใช้ในการทดลองมาผ่านกระบวนการตัดคำ

นำข้อความที่จัดเตรียมเอาไว้มาทำการตัดคำเพื่อที่จะนำไปใช้ในกระบวนการสกัด NE และการสกัดความสัมพันธ์ ซึ่งในขั้นตอนนี้จะใช้โปรแกรมตัดคำ KU Wordcut โดยข้อความที่จัดเตรียมไว้นั้นอยู่ในรูปแบบของไฟล์ข้อความ (text file) และก่อนที่จะนำไปผ่านกระบวนการตัดคำข้อความจะมีลักษณะดังนี้

> ไตรมาสที่ 2 (เมษา - พฤษภาฯ) ของปี 47 เมนบอร์ดของคอมพ์รุ่นใหม่จะมีชิปเซตสำหรับการเชื่อมต่อแบบไร้สาย (Wi-Fi) ติดตั้งไว้ด้วย ซึ่งจะทำให้โน้ตบุ๊ก Centrino สามารถเชื่อมต่อกับเครื่องเดสก์ทอปรุ่นใหม่แบบไร้สายได้ทันที โดยไม่ต้องซื้ออุปกรณ์เพิ่มแต่อย่างใด นอกจากนี้ ขนาดของเครื่องคอมพิวเตอร์ก็จะเล็กลงเนื่องจากมาตรฐานเมนบอร์ด BTX อีกด้วย

และเมื่อนำมาผ่านการตัดคำแล้วจะมีลักษณะดังนี้

> ไตรมาส ที่ _ 2 _ (เมษา _ - _ พฤษภา ฯ) _ ของ ปี _ 47 _ เมนบอร์ด ของ คอมพ์ รุ่น ใหม่ จะ มี ชิปเซต _ สำหรับ การ เชื่อมต่อ แบบ ไร้ สาย _ (Wi-Fi) _ ติดตั้ง ไว้ ด้วย _ ซึ่ง จะ ทำให้ โน้ตบุ๊ก _ Centrino _ สามารถ เชื่อมต่อ กับ เครื่อง เดสก์ทอป รุ่นใหม่ แบบ ไร้ สาย ได้ ทันที _ โดย ไม่ ต้อง ซื้อ อุปกรณ์ เพิ่ม แต่อย่างใด _ นอกจากนี้ _ ขนาด ของ เครื่อง คอมพิวเตอร์ ก็ จะ เล็ก ลง เนื่องจาก มาตรฐาน เมนบอร์ด _ BTX _ อีกด้วย

จากตัวอย่างผลลัพธ์จะเห็นว่าโปรแกรมจะแบ่งคำแต่ละคำออกจากกันด้วยการเว้นวรรค ส่วนการเว้นวรรคจากข้อความเดิมจะถูกแทนที่ด้วย “_” ซึ่งในส่วนนี้เราสามารถเลือกได้จากในโปรแกรมว่าจะให้ใช้เครื่องหมายใด (หากไม่กำหนดโปรแกรมจะใช้การเว้นบรรทัด)

3. สกัด NE

ทำการสกัด NE โดยจะใช้รายชื่อที่จัดเตรียมเอาไว้ในข้อ 1.1 เพื่อช่วยให้ระบบสามารถสกัด NE ออกมาได้ และใช้ชุดข้อมูลที่เตรียมเอาไว้ป้อนให้กับระบบทำการประมวลผล

ในขั้นตอนนี้จะใช้วิธีการในลักษณะเดียวกันกับของ Charoenpornsawat และคณะ (Charoenpornsawat, Kijsirikul and Maknavin 1998) ในการเลือกคำที่มีความเป็นไปได้ที่จะนำไปสกัดเป็น NE เช่น ในการสกัดชื่อเฉพาะในประโยค “นางเจนนีไปเดินตากลมเล่น” โดยที่คำที่เป็นตัวหนานั้นคือคำที่มีความเป็นไปได้ที่จะเป็นชื่อเฉพาะ โดยวิธีการคือ นำประโยคดังกล่าวไปผ่านกระบวนการตัดคำ และกำกับหน้าที่ของคำ ซึ่งจะได้ผลลัพธ์คือ “นาง/NTTL เจ/NCMN นนี/NPRP ไป/VACT เดิน/VACT ตาก/PPRS ลม/NCMN เล่น/ADVN” ซึ่งจะพบว่ามีคำซึ่งถูกกำกับหน้าที่ของคำเอาไว้ว่าเป็นชื่อเฉพาะ (/NPRP) โดยระบบจะนำคำที่ถูกกำกับหน้าที่ดังกล่าวมาเป็นคำเริ่มต้นที่จะนำไปค้นหา โดยจะทำการรวมคำที่อยู่ในระยะ ± 2 คำ (ดังเช่นในตารางที่ 5) และนำคำต่างๆ ไปตรวจสอบกับคลังข้อมูล

ตารางที่ 5 คำที่มีความเป็นไปได้ที่จะเป็นชื่อเฉพาะ (ตัวหนา)

นาง/t1 เจ/t2 **นนี/NPRP** ไป/t3 เดิน/t4 ตาก/t5 ลม/t6 เล่น/t7
นาง/t1 เจ/t2 **นนีไป/NPRP** เดิน/t3 ตาก/t4 ลม/t5 เล่น/t6
นาง/t1 เจ/t2 **นนีไปเดิน/NPRP** ตาก/t3 ลม/t4 เล่น/t5
นาง/t1 **เจนนี/NPRP** ไป/t2 เดิน/t3 ตาก/t4 ลม/t5 เล่น/t6
นาง/t1 **เจนนีไป/NPRP** เดิน/t2 ตาก/t3 ลม/t4 เล่น/t5
นาง/t1 **เจนนีไปเดิน/NPRP** ตาก/t2 ลม/t3 เล่น/t4
นางเจนนี/NPRP ไป/t1 เดิน/t2 ตาก/t3 ลม/t4 เล่น/t5
นางเจนนีไป/NPRP เดิน/t1 ตาก/t2 ลม/t3 เล่น/t4
นางเจนนีไปเดิน/NPRP ตาก/t1 ลม/t2 เล่น/t3

แต่จากการที่ไม่มีระบบกำกับหน้าที่ของคำ ดังนั้นในการสกัด NE ครั้งนี้จึงต้องมีการปรับปรุงวิธีการสกัดให้เข้ากับข้อมูลที่จะนำมาใช้ซึ่งเป็นลักษณะพจนานุกรมรายชื่อ โดยการเปลี่ยนมาเป็นการตรวจสอบคำตั้งแต่คำแรกในประโยค ยกตัวอย่างเช่นประโยคที่ว่า “มือถือรุ่นใหม่เป็นแบบพับได้” เมื่อผ่านการตัดคำแล้วจะได้ผลลัพธ์คือ “มือ ถือ รุ่นใหม่ เป็น แบบ พับ ได้” และเมื่อนำไปทำการสกัด NE ระบบจะตรวจสอบทีละคำจากคำที่ถูกแบ่งมาแล้วจากการตัดคำ เมื่อพบคำใดที่มีความเป็นไปได้ที่จะเป็น NE ระบบจะทำการผสมคำนั้นกับคำอื่นๆ ในบริบท เพื่อหาคำที่เป็นไปได้อื่นๆ (ดังเช่นในตารางที่ 6) และจะไม่ทำการผสมคำเพียงแค่ในระยะ ± 2 แต่จะหยุดผสมคำต่อเมื่อคำล่าสุดไม่สามารถค้นเจอได้ในพจนานุกรมรายชื่อ คำเป้าหมายจะถูกนำไปตรวจสอบในพจนานุกรมและหากพบคำใดที่ตรงกับที่มีอยู่ในพจนานุกรมที่สุดระบบจะเลือกคำนั้นมาเป็น NE

ตารางที่ 6 คำที่เป็นไปได้ที่ระบบค้นพบ (คำที่เป็นตัวหนาคือคำที่เป็นไปได้)

1. **มือ** ถือ รุ่นใหม่ เป็น แบบ พับ ได้
2. **มือ ถือ** รุ่นใหม่ เป็น แบบ พับ ได้
3. **มือ ถือ รุ่นใหม่** เป็น แบบ พับ ได้

จากตารางที่ 6 เมื่อนำคำที่เป็นไปได้ทุกคำไปตรวจสอบจากพจนานุกรมแล้วระบบจะเลือกข้อที่ 2 เป็น NE และกำกับว่าเป็น NE ประเภท PRO เนื่องจากคำในข้อที่ 3 นั้นไม่มีอยู่ในพจนานุกรมรายชื่อ

จากที่ได้กล่าวไปแล้วในเรื่องของปัญหาของคลังข้อมูลที่ใช้ในระบบนี้ ประกอบกับจำนวนรายชื่อที่นำมาใช้นั้นมีปริมาณที่ไม่มากนัก ดังนั้นผลลัพธ์ของการสกัด NE จึงอาจจะไม่มีความถูกต้องแม่นยำเพียงพอ ดังนั้นจึงได้เพิ่มวิธีการระบุขอบเขตของ NE โดยใช้กฎ เพื่อช่วยเพิ่มประสิทธิภาพในการสกัด NE โดยกฎที่นำมาใช้ในระบบนี้มีดังนี้

3.1. NE ที่ตามด้วยคำภาษาอังกฤษ

จากที่ได้ศึกษาข้อความภาษาไทย พบว่าการใช้ข้อความภาษาอังกฤษในภาษาไทย (ส่วนใหญ่)จะเป็นการใช้เพื่อกล่าวถึงชื่อสิ่งต่างๆ ที่เป็นชื่อภาษาอังกฤษ ตัวอย่างเช่น “เมนบอร์ด BTX” ในกรณีนี้หากในพจนานุกรมรายชื่อรู้จักแต่คำว่า “เมนบอร์ด” แต่ไม่รู้จักคำว่า BTX ซึ่งจะทำให้สกัด NE ไม่ถูกต้องหรือไม่ครบถ้วนได้

ดังนั้นจึงได้กำหนดให้เมื่อระบบตรวจพบ NE ในข้อความแล้วจะต้องตรวจสอบต่อไปด้วยว่าคำที่อยู่ถัดจาก NE นั้นเป็นคำภาษาอังกฤษหรือไม่

3.2. การสกัด NE ประเภท PER

เนื่องจากชื่อบุคคลในภาษาไทยนั้นมีหลายคำที่มีการสะกดแบบเดียวกับคำที่มีความหมายอื่นๆ ในภาษาไทย ยกตัวอย่างเช่น อยู่, แดง, ชม, น้อย, ยืน, นิด, อภัย, สามารถ เป็นต้น และเนื่องจากระบบนี้ไม่มีการใช้คลังข้อมูลที่สามารถใช้อ้างอิงสถิติการเกิดขึ้นของคำ จึงอาจทำให้การสกัด NE ประเภทนี้ไม่มีประสิทธิภาพ

ดังนั้นจึงได้กำหนดให้ระบบนี้จำเป็นที่จะต้องมีการตรวจสอบหาคำนำหน้าชื่อก่อน ซึ่งเป็นการพิจารณาลักษณะพิเศษของคำ (Chanlekha and Kawtrakul 2004) เมื่อพบแล้วจึงตรวจสอบหลังคำนำหน้าชื่อว่ามีคำที่สามารถจะเป็นชื่อบุคคลอยู่หรือไม่ หากพบว่ามีอยู่จึงให้กำหนดว่าเป็น NE ประเภท PER ยกเว้นแต่ว่าระบบตรวจพบเจอคำที่ไม่ใช่ชื่อบุคคลแต่คำนั้นมีความหมายอยู่ในประเภท PER ยกตัวอย่างเช่น นักเรียน, นักศึกษา, นักข่าว, คนไข้, ผู้ใช้ เป็นต้น หากเป็นคำที่ลักษณะเช่นนี้ระบบจะกำหนดให้คำนั้นเป็น NE ได้โดยที่ไม่จำเป็นต้องตรวจสอบหาคำนำหน้าชื่อก่อน

3.3. NE ที่มีตำแหน่งติดกัน

คำในภาษามนุษย์นั้นสามารถสร้างขึ้นได้จากคำหลายๆ คำมารวมกัน ในบางครั้งคำๆ หนึ่งอาจจะเกิดจากการรวมกันของคำเดี่ยวที่มีความหมายแตกต่างมารวมกันจึงเกิดเป็นคำที่มีความหมายต่างออกไป หรือเกิดเป็นคำที่มีความหมายสมบูรณ์ขึ้น

เช่นเดียวกับ NE ซึ่งเป็นคำนามชนิดหนึ่งที่สามารถเกิดขึ้นได้จากคำเดี่ยวหนึ่งคำ หรือเกิดจาก NE ตั้งแต่สองคำขึ้นไปเมื่อมาอยู่ติดกันแล้วจึงเกิดเป็น NE คำใหม่ขึ้นมาหนึ่งคำก็เป็นได้ ยกตัวอย่างดังเช่นตารางที่ 7

ตารางที่ 7 ตัวอย่าง NE ที่เกิดจาก NE ที่อยู่ติดกันมารวมกัน

NE ใหม่ที่เกิดจาก NE อื่นๆ มารวมกัน	**NE ที่ก่อให้เกิด NE ใหม่**
มาตรฐานเมนบอร์ด(PRO)	มาตรฐาน(PRO), เมนบอร์ด(PRO)
โปรแกรม Windows Media Player(PRO)	โปรแกรม(PRO), Windows Media Player(PRO)
การ์ดหน่วยความจำ SanDisk(PRO)	การ์ด(PRO), หน่วยความจำ(PRO), SanDisk(ORG)

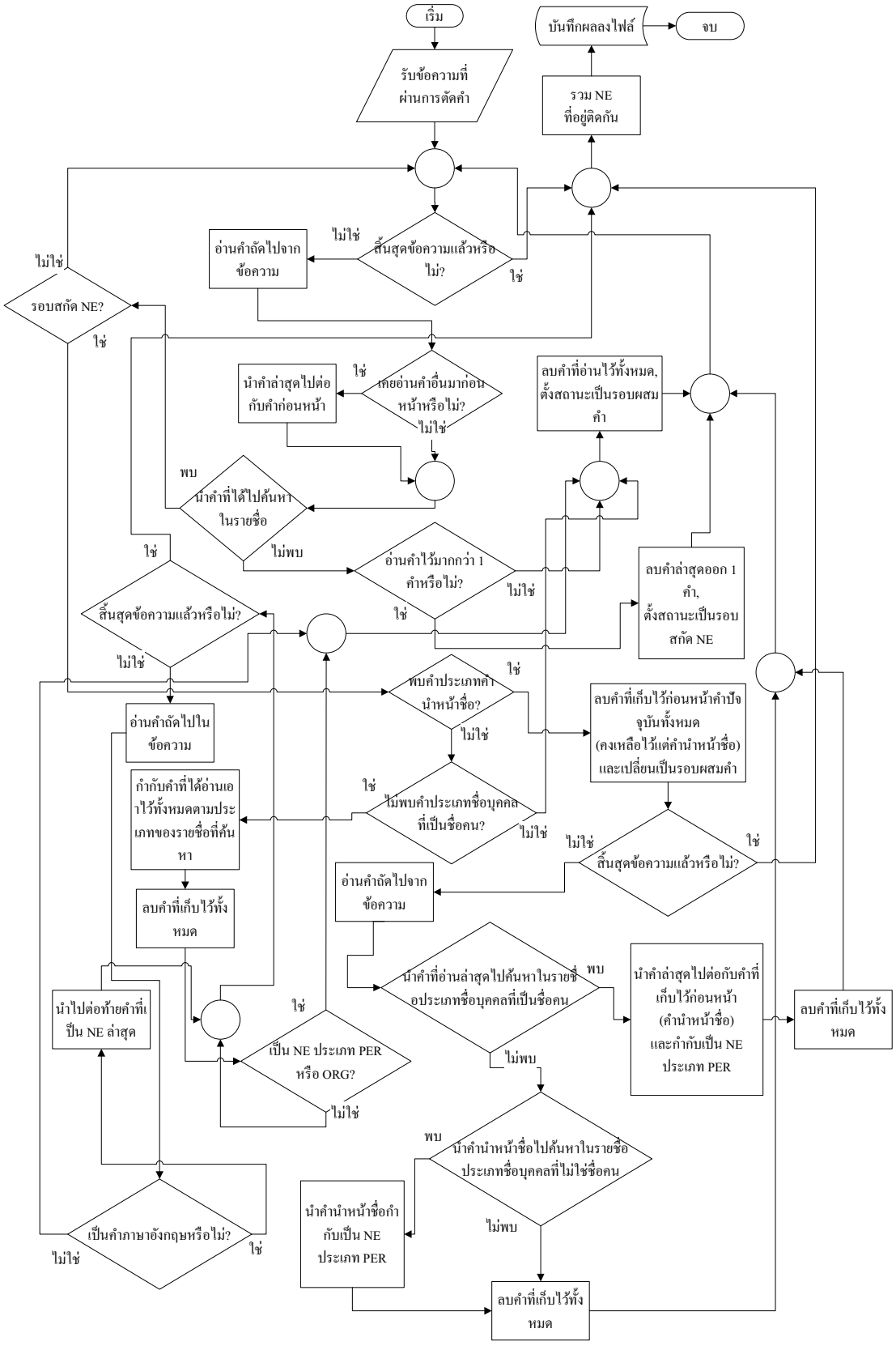
ดังนั้นในระบบนี้จึงกำหนดให้ เมื่อตรวจพบ NE จำนวนตั้งแต่สองคำขึ้นไปเกิดขึ้นในตำแหน่งติดกัน ให้รวม NE เหล่านั้นเป็น NE เพียงคำเดียว และกำหนดประเภทของ NE ใหม่ให้

เป็นประเภทเดียวกับ NE ตัวแรกที่ถูกรวมเข้าไป แต่ทั้งนี้มีเงื่อนไขบางประการที่จะทำให้ไม่เกิดการรวม NE ที่อยู่ติดกัน โดยเงื่อนไขเหล่านั้นมีดังนี้

- **NE แรกเป็น PER** เนื่องจากชื่อคนนั้นไม่สามารถนำไปผสมกับคำอื่นเพื่อให้เกิดคำใหม่ขึ้นมาได้ ดังนั้นหาก NE ตัวแรกเป็นประเภท PER จึงไม่ต้องนำไปรวมกับ NE ใดๆ
- **NE แรกเป็น ORG และ NE ถัดมาเป็น PRO** ถึงแม้จะมี NE ประเภท PRO บางคำที่ขึ้นต้นด้วยชื่อองค์กร เช่น Microsoft Windows เป็นต้น แต่ในข้อความภาษาไทยหลายๆ กรณีก็มีความเป็นไปได้ที่จะมีการกล่าวถึง NE ประเภท ORG และตามติดด้วย NE ประเภท PRO โดยที่ NE ทั้งสองไม่ได้มีความเกี่ยวข้องกัน เช่น “|PRO|ฮาร์ดดิสก์|/PRO| จิ๋ว _ จาก |ORG|โตชิบ้า|/ORG| _ |PRO|สินค้า|/PRO| เทคโนโลยี ยุค ใหม่ ที่ _ Hi-End _ Trendy _ เหล่านี้ _ คน ไอ ที เขา เรียก กัน ว่า _ |PRO|Gadget|/PRO|” (คำที่ขีดเส้นใต้คือ NE ที่อยู่ติดกัน) ดังนั้นผู้วิจัยจึงได้เลือกที่จะไม่รวม NE เหล่านี้ ด้วยเหตุที่ว่าไม่สามารถรับรองได้ว่าการรวม NE ประเภทนี้จะให้ผลลัพธ์ที่ถูกต้องทุกครั้ง และการเลือกที่จะไม่รวม NE ประเภทนี้ก็ให้ผลลัพธ์ที่สามารถยอมรับได้ โดยที่ความหมายของ NE ไม่เสียไปอีกด้วย
- **NE แรกเป็น LOC และ NE ถัดมาเป็น PRO** ด้วยเหตุผลเดียวกันกับข้อที่แล้ว (NE แรกเป็น ORG และถัดมาเป็น PRO) จึงเลือกที่จะไม่รวม NE ประเภทนี้

3.4. ขั้นตอนในการสกัด NE

ในส่วนของการสกัด NE มีขั้นตอนในการทำงานดังที่แสดงในภาพที่ 3



ภาพที่ 3 ผังแสดงขั้นตอนการสกัด NE

จากภาพที่ 3 แสดงขั้นตอนในการสกัด NE โดยในขั้นตอนแรกคือการรับข้อความที่ผ่านการตัดคำเข้ามาในระบบ จากนั้นจะตรวจสอบว่าได้ทำการตรวจสอบจนสิ้นสุดข้อความแล้วหรือไม่ (จุดนี้คือจุดเริ่มต้นของการตรวจสอบ NE มีลักษณะการทำงานแบบวนซ้ำ (Loop)) หากใช่จะออกจากการตรวจสอบและไปทำงานในส่วนของการรวม NE ที่อยู่ติดกัน หากไม่ใช่ให้อ่านคำถัดไปในข้อความ (นับจากตำแหน่งของตัวชี้ หรือ Pointer) จากนั้นจะตรวจสอบว่าทีคำอื่นใดที่เคยอ่านมาก่อนหน้าเก็บอยู่ในระบบหรือไม่ หากไม่ใช่ให้นำคำที่ได้ไปค้นหาในพจนานุกรมรายชื่อที่เตรียมไว้ หากใช่ให้นำคำล่าสุดไปต่อกับคำที่มีอยู่ก่อนหน้า จากนั้นจึงนำคำที่ได้ไปค้นหาในรายชื่อที่เตรียมไว้เช่นเดียวกัน หากไม่พบคำในรายชื่อให้ตรวจสอบว่ามีคำที่เก็บไว้ในระบบมากกว่า 1 คำหรือไม่ หากไม่ใช่ให้ลบคำที่เก็บเอาไว้ทั้งหมด และคำสถานะการทำงานให้เป็นรอบผสมคำ (สถานะการทำงานของระบบมีอยู่ 2 สถานะ คือ รอบผสมคำ ซึ่งจะเป็นรอบที่จะทำการผสมคำที่ได้จากข้อความ เพื่อสร้างคำที่เป็นไปได้ที่จะเป็น NE และในคำสั่งที่ใช้ตรวจสอบกับฐานข้อมูลจะใช้คำว่า “LIKE” ในการตรวจสอบ สำหรับอีกสถานะหนึ่ง คือ รอบสกัด NE ซึ่งจะไม่มีการผสมคำในรอบนี้ และในการตรวจสอบกับฐานข้อมูลจะใช้เครื่องหมายเท่ากับ “=” ในการตรวจสอบ) และไปเริ่มการทำงานที่จุดเริ่มต้นใหม่ แต่หากพบว่าระบบเก็บคำเอาไว้มากกว่าหนึ่งคำ ให้ลบคำที่เพิ่มเข้ามาล่าสุดออก 1 คำ และกลับไปเริ่มทำงานที่จุดเริ่มต้นเช่นเดียวกัน แต่หากตรวจสอบคำที่ได้และพบคำในรายชื่อให้ทำการตรวจสอบสถานะการทำงานว่าอยู่ในรอบสกัด NE หรือไม่ หากไม่ใช่ให้กลับไปทำงานที่จุดเริ่มต้น แต่หากใช่ให้ตรวจสอบว่าคำที่พบนั้นเป็นประเภทคำนำหน้าชื่อหรือไม่ หากใช่ให้ลบคำที่เก็บไว้ก่อนหน้าทั้งหมด คงเหลือไว้แต่คำนำหน้าชื่อที่พบ และเปลี่ยนสถานะการทำงานเป็นรอบผสมคำ จากนั้นทำการตรวจสอบว่าสิ้นสุดข้อความแล้วหรือไม่ หากใช่ให้ออกจากการตรวจสอบและไปทำงานในส่วนของการรวม NE ที่อยู่ติดกัน แต่หากไม่ใช่ให้อ่านคำถัดไปในข้อความ และนำคำนั้นไปค้นหาในรายชื่อประเภทชื่อคน หากพบให้นำคำล่าสุดไปต่อกับคำที่เก็บเอาไว้ (คำนำหน้าชื่อ) และกำกับคำในตำแหน่งนี้เป็นชนิด PER จากนั้นลบคำที่เก็บเอาไว้ทั้งหมดและกลับไปทำงานที่จุดเริ่มต้น แต่หากไม่พบให้นำคำนำหน้าชื่อไปค้นหาในรายชื่อประเภทบุคคลที่ไม่ใช่ชื่อคน หากไม่พบให้ลบคำที่เก็บเอาไว้ทั้งหมดและกลับไปทำงานที่จุดเริ่มต้น แต่หากพบให้กำกับคำในตำแหน่งนี้เป็น NE ชนิด PER จากนั้นลบคำที่เก็บไว้ทั้งหมดและกลับไปทำงานที่จุดเริ่มต้น แต่หากว่าไม่ได้พบคำนำหน้าชื่อ ให้ตรวจสอบว่าคำที่พบนั้นไม่ได้เป็นชื่อคนใช่หรือไม่ หากไม่ใช่ให้ลบคำที่เก็บไว้ทั้งหมด และเปลี่ยนสถานะเป็นรอบผสมคำ และกลับไปทำงานที่จุดเริ่มต้น แต่หากใช่ให้กำกับคำที่เก็บไว้เป็น NE ตามประเภทของรายชื่อที่พบคำนั้น และลบคำที่ได้เก็บไว้ทั้งหมด จากนั้นตรวจสอบประเภทของคำว่าเป็นประเภท ORG หรือ PER ใช่หรือไม่ หากใช่ให้เปลี่ยนเป็นรอบผสมคำและกลับไปเริ่มทำงานที่จุดเริ่มต้น แต่หากไม่ใช่ให้ตรวจสอบว่าสิ้นสุด

ข้อความแล้วหรือไม่ หากใช่ให้ออกจากการตรวจสอบและไปทำงานในส่วนของการรวม NE ที่อยู่ติดกัน แต่หากไม่ใช่ให้อ่านคำถัดไปในข้อความและตรวจสอบว่าเป็นคำภาษาอังกฤษหรือไม่ หากใช่ให้นำคำไปต่อท้ายคำที่เป็น NE คำล่าสุด และกลับไปทำซ้ำในขั้นตอนการตรวจสอบการสิ้นสุดข้อความไปจนถึงตรวจสอบคำภาษาอังกฤษอีกครั้ง จนกว่าจะไม่พบคำภาษาอังกฤษอีก ก็จะออกจากส่วนนี้ และทำการลบคำที่ได้เก็บไว้ทั้งหมด และเปลี่ยนสถานะเป็นรอบผสมคำ จากนั้นจึงกลับไปทำงานที่จุดเริ่มต้น ระบบจำทำเช่นนี้ไปจนกว่าจะตรวจสอบข้อความจนสิ้นสุดข้อความก็จะออกจากการตรวจสอบ NE ในข้อความ และเข้าสู่การรวม NE ที่มีตำแหน่งติดกัน โดยในส่วนนี้จะรวม NE ที่มีตำแหน่งติดกันเข้าด้วยกันตามกฎที่ได้ตั้งเอาไว้ เมื่อทำขั้นตอนในส่วนนี้เสร็จแล้ว ระบบก็จะบันทึกผลลัพธ์ลงในไฟล์ข้อความ จึงเป็นอันเสร็จสิ้นกระบวนการสกัด NE

4. สกัดความสัมพันธ์

พัฒนาระบบสกัดความสัมพันธ์ระหว่าง NE โดยนำแนวคิดของ Hasegawa และคณะ (Hasegawa, Sekine and Grishman 2004) ที่ว่าด้วยการนำบริบทที่อยู่ระหว่าง NE สองคำ (NE1…NE2) มาทำการวิเคราะห์หาความสัมพันธ์ ตัวอย่างเช่น “เมนบอร์ด ของ คอมพ์ รุ่นใหม่ จะ มี ชิปเซต” จากประโยคนี้เมื่อผ่านการกำกับ NE แล้ว จะได้เป็น “|PRO|เมนบอร์ด|/PRO| ของ คอมพ์ รุ่นใหม่ จะ มี |PRO|ชิปเซต|/PRO|” จากนั้นจะทำการสกัดความสัมพันธ์ ซึ่งในส่วนนี้ข้อความที่อยู่ระหว่าง NE ทั้งสองนั้นเรียกว่าบริบท (Context) และยังได้นำแนวคิดของ Appelt และคณะ (Appelt et al. 1993) ในการค้นหาคำที่สื่อถึงความหมาย (Trigger Word) ภายในข้อความ โดยจะนำวิธีการแบบ Regular Expression และ Heuristic มาใช้ โดยการค้นหาคำในบริบทที่มีความหมายแสดงถึงความสัมพันธ์ โดยนำคำแต่ละคำในบริบทไปทำการค้นหาในรายชื่อคำสำคัญที่ได้จัดเตรียมเอาไว้ (ในข้อ 1.3) ซึ่งคำสำคัญนั้นสามารถที่จะบ่งบอกถึงความสัมพันธ์ในประโยคได้ โดยในการค้นหานั้นระบบจะเริ่มค้นหาโดยการค้นหาคำสำคัญที่เป็นกริยาก่อน หากพบคำสำคัญในบริบทจะเริ่มขั้นตอนการตรวจสอบคุณสมบัติในบริบทนั้นๆ ว่าตรงกับที่กำหนดไว้ในคำสำคัญที่ค้นพบหรือไม่ แต่หากไม่พบคำสำคัญที่เป็นคำกริยาในบริบทนั้น หรือพบแต่คุณสมบัติไม่ตรงกับที่กำหนดไว้ ระบบจะทำการค้นหาจากคำสำคัญที่เป็นคำบุพบทและดำเนินการตามขั้นตอนเดียวกันกับคำกริยา

จากในตัวอย่าง (ดูภาพที่ 4 ประกอบ) คำสำคัญที่ค้นพบก็คือคำว่า “มี” (คำกริยา) ซึ่งสื่อให้รู้ว่ามีสิ่งของบางอย่างติดตั้งอยู่ในสิ่งของอีกหนึ่งอย่าง และเมื่อผ่านการตรวจสอบคุณสมบัติแล้ว

พบว่าบริบทนี้มีคุณสมบัติตรงกับที่ได้กำหนดไว้ให้กับคำว่า “มี” ดังนั้นจากประโยคนี้จึงกล่าวได้ว่า NE คำแรกมีความสัมพันธ์กับ NE คำที่สองแบบ located_in

ภาพที่ 4 แสดงลักษณะการสกัดความสัมพันธ์โดยการใช้คำสำคัญ

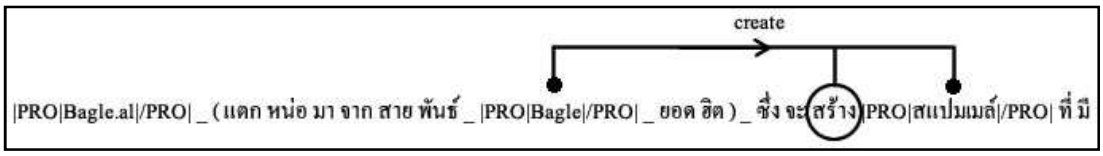
ในขั้นตอนของการสกัดความสัมพันธ์ นอกเหนือจากการใช้คำสำคัญในการตรวจสอบความสัมพันธ์แล้ว ยังมีปัญหาอื่นซึ่งเกิดจากลักษณะการเขียนข้อความ หรือลักษณะการใช้ภาษา ซึ่งมีความหลากหลายในการใช้เป็นอย่างมาก จึงทำให้การใช้วิธีค้นหาคำสำคัญที่บ่งบอกถึงความสัมพันธ์ในข้อความเพียงอย่างเดียวนั้นไม่สามารถที่จะสกัดความสัมพันธ์ออกมาได้ถูกต้องเสมอไป เพราะยังมีปัจจัยอื่นๆ ที่อาจจะทำให้ระบบเกิดความเข้าใจผิดได้ ดังนั้นระบบจึงจำเป็นที่จะต้องมีความเข้าใจลักษณะการใช้ภาษาแบบต่างๆ และต่อไปนี้คือสิ่งที่จำเป็นจะต้องคำนึงถึงเมื่อต้องการที่จะสกัดความสัมพันธ์

4.1. คุณลักษณะของข้อความที่จะนำมาสกัด

4.1.1. ข้อความที่มีเครื่องหมายวงเล็บ

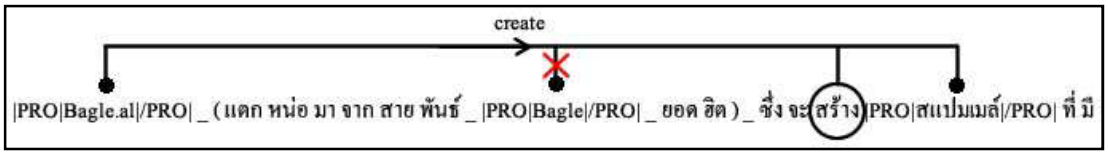
โดยปกติเมื่อในข้อความมีการใช้เครื่องหมายวงเล็บครอบข้อความใดข้อความหนึ่งไว้ ข้อความนั้นมักจะหมายถึงการขยายความข้อความที่อยู่ภายนอกวงเล็บ ยกตัวอย่างเช่น “ไวรัสตัวใหม่นี้ชื่อว่า Bagle.al (แตกหน่อมาจากสายพันธ์ Bagle ยอดฮิต) ซึ่งจะสร้างสแปมเมล์ที่มีเนื้อความง่ายๆ” ซึ่งข้อความในวงเล็บนั้นใช้ขยายความว่า ไวรัสชื่อ Bagle.al มาจากสายพันธุ์ Bagle และเมื่อผ่านการกำกับ NE แล้วจะเป็นดังนี้ “|PRO|ไวรัส|/PRO| ตัว ใหม่ นี้ ชื่อ ว่า _ |PRO|Bagle.al|/PRO| _ (แตก หน่อ มา จาก สาย พันธ์ _ |PRO|Bagle|/PRO| _ ยอด ฮิต) _ ซึ่ง จะ สร้าง |PRO|สแปมเมล์|/PRO| ที่ มี |PRO|เนื้อความ|/PRO| ง่าย ๆ” และเมื่อผ่านการวิเคราะห์หาความสัมพันธ์แล้ว จะพบว่าข้อความนี้มีความสัมพันธ์ที่ว่า “ไวรัสชื่อ Bagle.al สร้างสแปมเมล์ขึ้นมา (|PRO|Bagle.al|/PRO| create |PRO|สแปมเมล์|/PRO|)” แต่ถ้าหากว่าระบบไม่สามารถเข้าใจว่า

ข้อความในวงเล็บนั้นเป็นเพียงส่วนขยายข้อความหลักเท่านั้น จะทำให้ระบบไปตรวจจับความสัมพันธ์ว่า “สายพันธุ์ Bagle สร้างสแปมเมล์ขึ้นมา (|PRO|Bagle|/PRO| created |PRO|สแปมเมล์|/PRO|)” ซึ่งไม่ถูกต้องตามความหมายที่ข้อความต้องการจะสื่อ นั่นเป็นเพราะว่าระบบจะทำการหาความสัมพันธ์ในข้อความระหว่าง NE ที่อยู่ใกล้ที่สุด ดังภาพที่ 5



ภาพที่ 5 การสกัดความสัมพันธ์ในกรณีที่ระบบไม่รู้จักวงเล็บ

ดังนั้นจึงกำหนดให้ระบบไม่ทำการสกัดความสัมพันธ์ระหว่าง NE ที่อยู่ในวงเล็บกับนอกวงเล็บ (แต่หากเป็นในวงเล็บด้วยกันอนุญาตให้มีความสัมพันธ์ได้) ดังภาพที่ 6



ภาพที่ 6 การสกัดความสัมพันธ์เมื่อระบบไม่สนใจข้อความในวงเล็บ

4.1.2. NE ที่มีความสัมพันธ์กันในลักษณะของความเป็นเจ้าของ

NE ที่มีความสัมพันธ์ในลักษณะนี้จะถูกใช้ในลักษณะของการขยายความหมายของ NE ในการบอกที่มาที่ไปของ NE ยกตัวอย่างเช่น “|PRO|ช่อง โหว่|/PRO| สำหรับ |PRO|ความ ปลอดภัย|/PRO|” จากตัวอย่างนี้จะเห็นว่าความหมายของประโยคนี้คือ “|PRO|ช่อง โหว่|/PRO| เป็นส่วนหนึ่งของ |PRO|ความปลอดภัย|/PRO|” ดังนั้นเมื่อนำ NE ลักษณะนี้ไปอยู่ในข้อความอย่างเช่น “|PRO|ช่อง โหว่|/PRO| สำหรับ |PRO|ความ ปลอดภัย|/PRO| ที่ พบ ใน _ |PRO|IE|/PRO|” เมื่อพิจารณาจากข้อความที่เพิ่มเข้ามาแล้วจะเห็นว่ามีความสัมพันธ์อยู่ในข้อความซึ่งความสัมพันธ์นั้นคือ “มี|PRO|ช่อง โหว่|/PRO| อยู่ใน |PRO|IE|/PRO|”

ปัญหาของการวิเคราะห์ข้อความที่มีความสัมพันธ์ของ NE ลักษณะนี้ก็คือ หากระบบทำการวิเคราะห์ความสัมพันธ์ตามปกติ ระบบจะทำการสกัดความสัมพันธ์ระหว่าง |PRO| ความปลอดภัย|/PRO| กับ |PRO|IE|/PRO| ดังภาพที่ 7

ภาพที่ 7 การสกัดความสัมพันธ์เมื่อระบบไม่รู้ว่า NE ด้านซ้ายมีความสัมพันธ์กับ NE อื่นอยู่ (ในลักษณะของความเป็นเจ้าของ)

ซึ่งไม่ใช่ความสัมพันธ์ที่ถูกต้องเนื่องจากระบบทำการจับคู่ NE ที่มีความสัมพันธ์ผิด ดังนั้นระบบจึงจำเป็นที่จะต้องมีความสามารถที่จะรู้ได้ว่า NE ทางด้านซ้ายนั้นมีความสัมพันธ์กับ NE ก่อนหน้าในลักษณะเป็นเจ้าของหรือไม่ โดยกำหนดให้ระบบต้องทำการตรวจสอบคำสำคัญที่บ่งบอกถึงความเป็นเจ้าของ อย่างเช่นในตัวอย่างนี้ใช้คำว่า “ของ” หรืออาจจะเป็นคำอื่นๆ เช่น จาก, เพื่อ, บน, ที่, สำหรับ เป็นต้น นอกจากนี้ในบริบทที่พบคำเหล่านี้จะต้องมีพบคำที่ไม่ต้องการให้เกิดร่วมกัน (ถ้ามี) (ดูตัวอย่างคำที่แสดงถึงความเป็นเจ้าของได้ที่หน้า 96) และคำเหล่านี้จะต้องอยู่ห่างจาก NE ทางซ้ายและขวาไม่เกิน 2 คำด้วย เนื่องจากคำเหล่านี้สามารถนำไปผสมกับคำอื่นๆ ที่ให้ความหมายต่างออกไปได้ และนอกจากการตรวจสอบหาคำสำคัญที่ทำให้เกิดความสัมพันธ์แบบความเป็นเจ้าของแล้ว ยังรวมไปถึงการที่ NE สองคำอยู่ติดกันโดยไม่มีคำใดๆ มาคั่นอีกด้วย ยกตัวอย่างเช่น “|ORG| บริษัท ผู้ ผลิต|/ORG| |PRO|การ์ด หน่วย ความจำ _ SanDisk|/PRO|”

จากตัวอย่างที่ได้กล่าวมานั้น หากระบบสามารถที่จะรู้ได้ว่า NE ทางซ้ายมือนั้นมีความสัมพันธ์กับ NE ก่อนหน้าในลักษณะของความเป็นเจ้าของ จะทำให้ระบบสามารถที่จะจับคู่ NE ที่มีความสัมพันธ์กันได้อย่างถูกต้อง ดังภาพที่ 8

ภาพที่ 8 การสกัด NE เมื่อระบบพบว่า NE ทางซ้ายมีความสัมพันธ์กับ NE อื่นอยู่ (ในลักษณะของความเป็นเจ้าของ)

4.1.3. ประโยคปฏิเสธ

"|PRO|ไฟล์|/PRO| ที่ ไม่ อยู่ ใน |PRO|เครื่อง พีซี|/PRO|" จากประโยคดังกล่าวจะเห็นว่าเป็นประโยคปฏิเสธ ซึ่งหากระบบทำการสกัดความสัมพันธ์ด้วยวิธีการค้นหาคำสำคัญที่บ่งบอกถึงความสัมพันธ์เพียงอย่างเดียว จะทำให้ระบบทำการกำกับความสัมพันธ์ในประโยคนี้ เนื่องจากได้พบคำว่า "อยู่" ประกอบกับคำว่า "ใน ซึ่งเป็นคำที่เกิดขึ้นร่วมกัน ซึ่งจะทำให้ระบบเข้าใจว่าประโยคนี้มีความสัมพันธ์แบบ located_in ซึ่งในความเป็นจริงแล้วจะต้องไม่มีความสัมพันธ์เกิดขึ้นในประโยคนี้ เนื่องจากเป็นประโยคปฏิเสธ

ดังนั้นระบบจึงจำเป็นที่จะต้องสามารถรับรู้ได้ว่าประโยคใดคือประโยคปฏิเสธ โดยการค้นหาคำที่บ่งบอกถึงการปฏิเสธ เช่นคำว่า ไม่, มิได้, มิใช่ เป็นต้น แต่ทั้งนี้คำเหล่านี้จะต้องไม่เกิดขึ้นร่วมกับคำบางคำที่ได้กำหนดไว้ เช่น คำว่า "ไม่" จะต้องไม่เกิดร่วมกับคำว่า "เพียง" (ไม่เพียง) หรือคำว่า "ว่า" (ไม่ว่า) เป็นต้น เมื่อค้นพบและได้ทำการทดสอบแล้วว่าเป็นประโยคปฏิเสธจริง ระบบจะไม่สนใจความสัมพันธ์ที่มีอยู่ในประโยคนั้น (ดูคำที่แสดงถึงประโยคปฏิเสธได้ที่หน้า 97)

4.2. การระบุขอบเขตของความสัมพันธ์

ในหัวข้อนี้จะกล่าวถึงขั้นตอนการสกัดความสัมพันธ์ที่น่าสนใจนอกเหนือจากที่ได้กล่าวไปแล้วในตอนต้น ซึ่งรวมไปถึงการสกัดความสัมพันธ์จากข้อความที่มีความสัมพันธ์ในลักษณะที่ได้กล่าวไป คือความสัมพันธ์ที่เกิดขึ้นจากบริบทที่อยู่ระหว่าง NE 2 คำที่มีความสัมพันธ์กัน (ดังเช่นในภาพที่ 4) แต่ในที่นี้จะกล่าวถึงความสัมพันธ์ที่มีขอบเขตกว้างมากกว่าที่กล่าวมา เช่น ความสัมพันธ์อาจจะไม่ได้เกิดขึ้นกับ NE คำที่อยู่ในบริบทที่พบคำสำคัญ หากแต่อยู่ห่างออกไปอีก

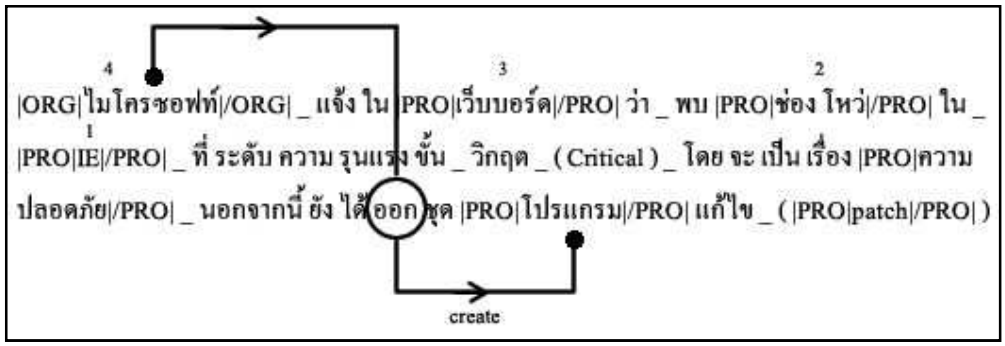
หลายบริบท ระบบจึงจำเป็นที่จะสามารถระบุขอบเขตของความสัมพันธ์ได้ ซึ่งวิธีการระบุขอบเขตมีดังนี้

4.2.1. ความสัมพันธ์ที่มีการกำหนดประเภทของ NE ทางด้านซ้าย

คำสำคัญที่มีการกำหนดประเภทของ NE นั้นหมายความว่า NE ที่อยู่รอบบริบทนั้น (ไม่ว่าซ้ายหรือขวา) จะต้องเป็นประเภทเดียวกับที่กำหนดเอาไว้ หากไม่ใช่จะถือว่ามีคุณสมบัติไม่ครบถ้วนตามที่คำสำคัญนั้นๆ ต้องการ แต่หากเป็น NE ที่อยู่ทางด้านซ้ายซึ่งไม่ตรงกับที่กำหนดเอาไว้ ระบบจะทำการค้นหาประเภท NE ที่ตรงกับประเภทที่ต้องการย้อนกลับไปยังต้นข้อความเป็นระยะทางไม่เกิน NE 4 คำ (เนื่องมาจากการที่ได้ทดลองกับชุดข้อความที่นำมาทดสอบพบว่า ระยะ 4 คำเป็นระยะที่ครอบคลุมถึง NE ที่มีลักษณะเช่นนี้ที่สุด) ยกตัวอย่างเช่น

> |ORG|ไมโครซอฟท์|/ORG| _ แจ้ง ใน |PRO|เว็บบอร์ด|/PRO| ว่า _ พบ |PRO|ช่อง โหว่|/PRO| ใน _ |PRO|IE|/PRO| _ ที่ ระดับ ความ รุนแรง ขั้น _ วิกฤต _ (Critical) _ โดย จะ เป็น เรื่อง |PRO|ความปลอดภัย|/PRO| _ นอกจากนี้ ยัง ได้ ออก ชุด |PRO|โปรแกรม|/PRO| แก้ไข _ (|PRO|patch|/PRO|)

จากตัวอย่างนี้โปรดสังเกตบริบทที่ขีดเส้นใต้ ซึ่งเป็นบริบทที่มีคำว่า “ออก” ซึ่งเป็นคำที่สื่อถึงความสันพันธ์ประเภท create แต่คำนี้ถูกกำหนดเอาไว้ว่าประเภทของ NE ทางซ้ายจะต้องเป็น ORG และทางขวาจะต้องเป็น PRO จากตัวอย่างจะเห็นว่า NE ทางด้านขวานั้นตรงกับที่ได้กำหนดไว้ แต่ NE ทางด้านซ้ายนั้นไม่ตรงกับที่กำหนด ดังนั้นหากระบบหยุดการทดสอบเพียงเท่านี้ บริบทนี้จะไม่มีความสัมพันธ์เกิดขึ้น แต่ในความเป็นจริงแล้วความสัมพันธ์ไม่จำเป็นจะต้องเกิดขึ้นกับ NE ที่อยู่ใกล้ที่สุดเสมอไป เพราะ NE คำที่มีความสัมพันธ์จริงๆ อาจจะถูกกล่าวถึงมาก่อนแล้ว และเมื่อกล่าวมาถึงบริบทที่มีความสัมพันธ์จริงๆ จึงไม่จำเป็นที่จะต้องกล่าวซ้ำอีก แต่ทั้งนี้จะต้องเป็นความสัมพันธ์ที่มีความน่าจะเป็นที่จะเกิดขึ้นด้วย ดังเช่นในตัวอย่าง เราพบคำว่า “ออก” ซึ่งสื่อถึงการผลิต แต่เมื่อพิจารณาถึง NE ที่เกิดขึ้นมาก่อนหน้าแล้วไม่มีความเป็นไปได้ที่จะเกิดความสัมพันธ์เช่นนี้ แต่หากพิจารณาย้อนกลับไปยัง NE ก่อนหน้าแล้ว (ไม่เกิน 4 คำ) พบว่า NE “|ORG|ไมโครซอฟท์|/ORG|” มีความเป็นไปได้ที่จะเป็นผู้ผลิต ดังนั้นในข้อความนี้จึงกล่าวได้ว่า |ORG|ไมโครซอฟท์|/ORG| มีความสัมพันธ์กับ |PRO|โปรแกรม|/PRO| แบบ create (ดังเช่นในภาพที่ 9)

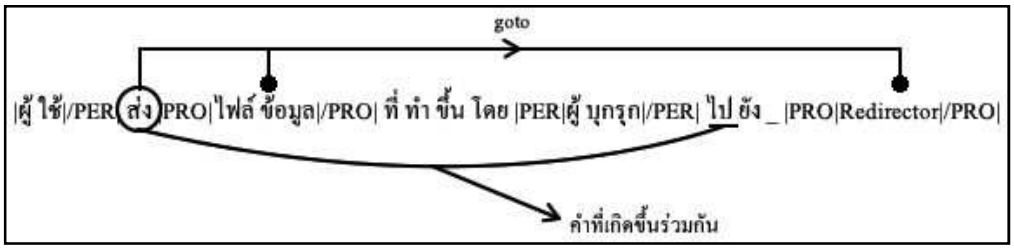


ภาพที่ 9 แสดงการสกัดความสัมพันธ์ในกรณีที่ NE ทางซ้ายไม่ตรงกับที่กำหนดไว้

4.2.2. ความสัมพันธ์ที่เกิดขึ้นในบริบทหลังจากบริบทปัจจุบัน

ความสัมพันธ์ในลักษณะนี้จะเกิดขึ้นในบริบทที่อยู่หลังจากบริบทที่พบความสัมพันธ์ ดังเช่นที่ได้กล่าวไว้ในหัวข้อที่ 1.3 (ncr_want) แต่ในบางครั้งความสัมพันธ์ลักษณะนี้อาจจะไม่ได้เกิดขึ้นในบริบทที่อยู่ติดกับบริบทปัจจุบันเท่านั้น คืออาจจะอยู่ห่างออกไปมากกว่าหนึ่งบริบท เช่น “|PER|ผู้ ใช้|/PER| ส่ง |PRO|ไฟล์ ข้อมูล|/PRO| ที่ ทำ ขึ้น โดย |PER|ผู้ บุกรุก|/PER| ไปยัง _ |PRO|Redirector|/PRO|” จากตัวอย่างให้สังเกตที่บริบทที่ถูกขีดเส้นใต้ซึ่งมีคำว่า “ส่ง” ซึ่งเป็นคำที่สื่อถึงความสัมพันธ์ประเภท goto โดยที่คำนี้ต้องการให้ในบริบทถัดไปมีคำว่า “ให้” หรือ “ไป” (ระบุไว้ใน want_after_ne2) คำใดคำหนึ่งจึงจะเกิดความสัมพันธ์ แต่เมื่อพิจารณาจากบริบทที่อยู่ถัดไปไม่ปรากฏคำดังกล่าว แต่คำดังกล่าวนั้นกลับไปปรากฏอยู่ในบริบทหลังจากนั้น และความสัมพันธ์ก็ยังคงอยู่

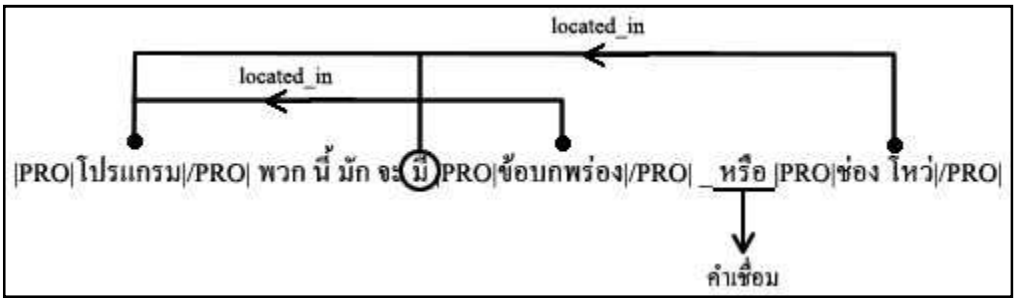
ดังนั้นจึงอาจกล่าวได้ว่าความสัมพันธ์บางประเภทนั้นสามารถเกิดขึ้นได้หลังจากมีคำสำคัญปรากฏขึ้นมา และความสัมพันธ์นั้นไม่จำเป็นที่จะต้องเกิดขึ้นทันทีที่บริบทถัดจากบริบทที่มีความสัมพันธ์ปรากฏขึ้นมา เนื่องจากในการใช้ภาษานั้นมีความเป็นไปได้ที่จะมีการกล่าวถึงความสัมพันธ์ของสิ่งใดสิ่งหนึ่งขึ้นมาก่อนโดยที่ความหมายของความสัมพันธ์นั้นยังไม่จบลงไป และตามด้วยการกล่าวถึงสิ่งอื่นขึ้นมาแทรกระหว่างความสัมพันธ์ (เพื่อเป็นการขยายความ) แล้วจากนั้นจึงกลับมากล่าวถึงเรื่องเดิมเพื่อเป็นการจบความหมายที่ต้องการจะสื่อ ดังเช่นตัวอย่างที่ได้กล่าวมา ดังนั้นระบบจึงจำเป็นที่จะต้องสามารถรับรู้ได้ว่าขอบเขตของความสัมพันธ์ที่พบนั้นจะไปสิ้นสุดที่บริบทใดจึงจะสามารถสกัดความสัมพันธ์ได้อย่างถูกต้อง (เช่นในภาพที่ 10)



ภาพที่ 10 แสดงการสกัดความสัมพันธ์เมื่อมีความสัมพันธ์เกิดขึ้นหลังจากบริบทปัจจุบันและความสัมพันธ์นั้นถูกแทรกด้วยข้อความอื่น

4.2.3. ความสัมพันธ์ที่เกิดขึ้นกับ NE จำนวนหลายคำ

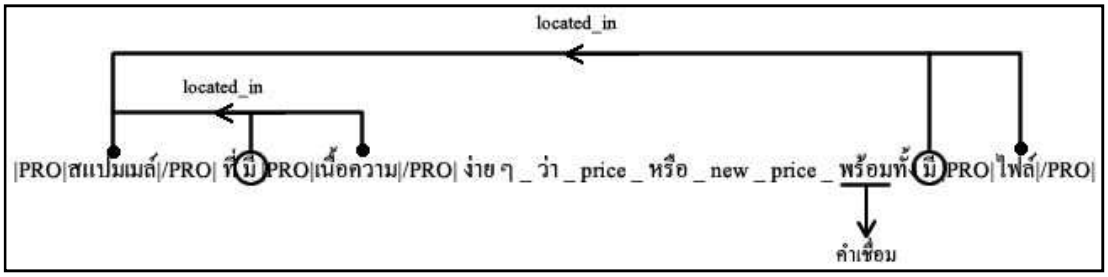
- ในข้อความใดเมื่อมีความสัมพันธ์เกิดขึ้นและ NE ทางด้านขวาของความสัมพันธ์ถูกเชื่อมกับ NE คำอื่นๆ ด้วยคำว่า ตั้งแต่, พร้อมกับ, หรือ, และ, ไปจนถึง รวมไปถึงเครื่องหมาย “,” เป็นต้น นั้นให้ถือว่า NE ที่อยู่ต่อจากคำเหล่านี้มีความสัมพันธ์เช่นเดียวกันกับ NE ก่อนหน้า ดังภาพที่ 11



ภาพที่ 11 แสดงการสกัดความสัมพันธ์เมื่อมีคำเชื่อม NE

- ในบริบทใดที่มีความสัมพันธ์และในบริบทนั้นมีคำว่า “เพื่อ” หรือ “พร้อม” เกิดขึ้นก่อนหน้าคำสำคัญ ให้ NE ทางขวาของบริบทนี้มีความสัมพันธ์กับ NE ทางซ้ายของบริบทก่อนหน้า ดังภาพที่ 12 และภาพที่ 13

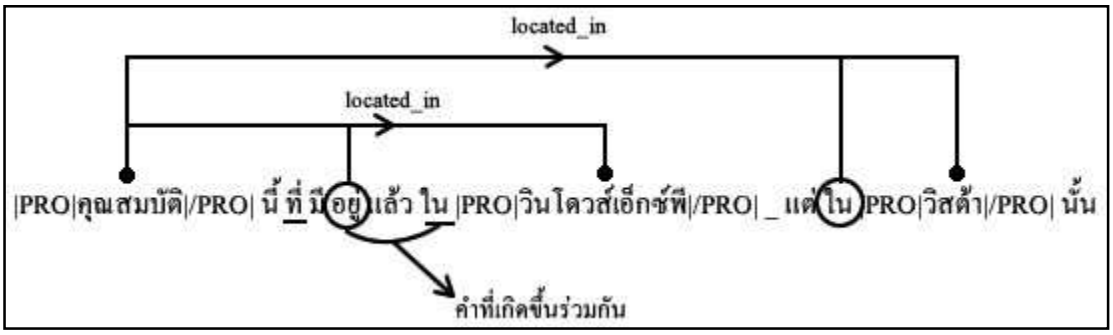
ภาพที่ 12 การสกัดความสัมพันธ์ในบริบทที่มีคำว่า “เพื่อ”



ภาพที่ 13 การสกัดความสัมพันธ์ในบริบทที่มีคำว่า “พร้อม”

เนื่องจากการใช้คำว่า “เพื่อ” หรือ “พร้อม” นั้นส่วนใหญ่จะสื่อถึงการมีเหตุการณ์เกิดขึ้นมาก่อนหน้า และเมื่อมาถึงบริบทที่มีคำนี้จะหมายถึงผลของการกระทำที่ได้เกิดขึ้นมาก่อนหน้า ดังนั้นหากมีความสัมพันธ์เกิดขึ้นในบริบทที่มีคำเหล่านี้จึงกำหนดให้ NE ทางขวาของบริบทนี้มีความสัมพันธ์กับ NE ทางซ้ายของบริบทที่เกิดขึ้นมาก่อนหน้า

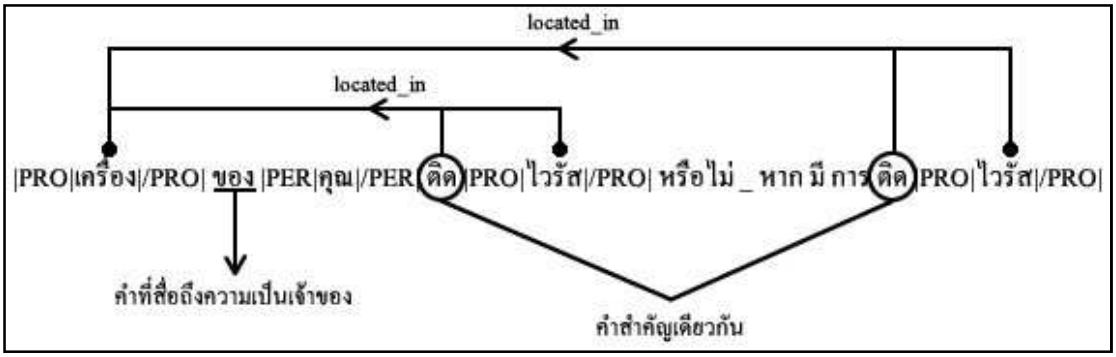
- หากในบริบทใดมีความสัมพันธ์เกิดขึ้นและในบริบทนั้นมีคำว่า “ที่” หรือ “ซึ่ง” อยู่ก่อนหน้าคำสำคัญในบริบทนั้น และเมื่อถึงบริบทถัดไปและในบริบทนั้นมีความสัมพันธ์เกิดขึ้น และในบริบทนั้นไม่มีคำว่า “ที่” หรือ “ซึ่ง” อยู่ ให้ NE ทางขวาของบริบทนี้มีความสัมพันธ์กับ NE ทางซ้ายกับบริบทแรก ดังภาพที่ 14



ภาพที่ 14 แสดงการสกัดความสัมพันธ์ในบริบทที่มีคำว่า “ที่” หรือ “ซึ่ง”

เนื่องจากคำว่า “ที่” หรือ “ซึ่ง” นั้นเป็นคำที่ให้ความรู้สึกเน้นความสำคัญของ NE ในบริบทนั้น และเมื่อในบริบทถัดไปเกิดความสัมพันธ์ขึ้นมา NE ในบริบทก่อนหน้าจึงยังมีความสัมพันธ์ต่อเนื่องมายังบริบทถัดมาด้วย

- หากในบริบทใดที่ NE ทางซ้ายเคยมีความสัมพันธ์มาแล้วในบริบทก่อนหน้า และในบริบทนี้มีความสัมพันธ์เกิดขึ้น ให้ตรวจสอบว่าในบริบทนี้ใช้คำสำคัญเดียวกันกับบริบทก่อนหน้าหรือไม่ หากใช่ให้ NE ทางขวาในบริบทนี้มีความสัมพันธ์กับ NE ทางซ้ายในบริบทก่อนหน้า ดังเช่นในรูปที่ 15

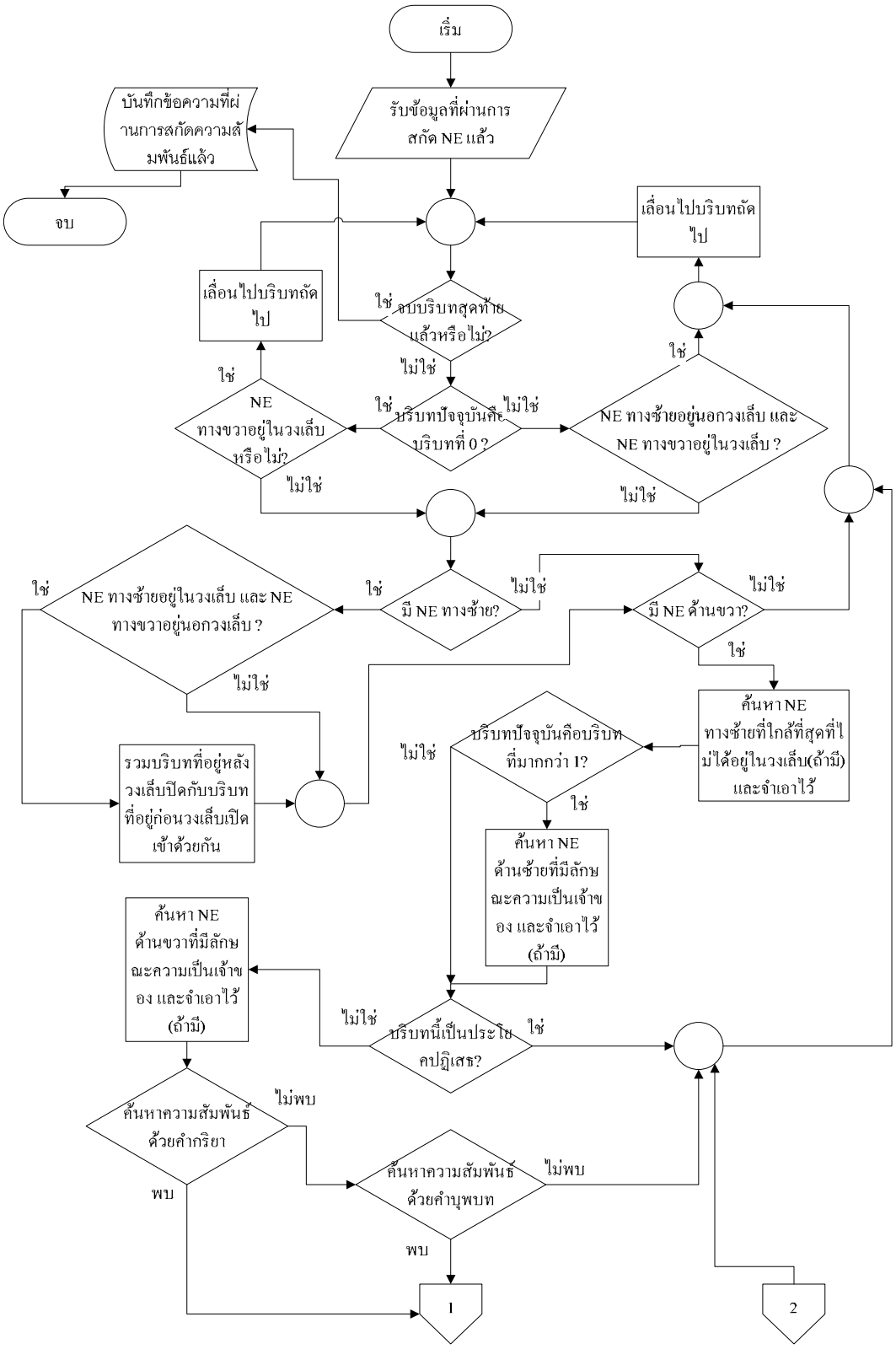


ภาพที่ 15 แสดงการสกัดความสัมพันธ์ในบริบทที่มีคำสำคัญเดียวกันกับบริบทก่อนหน้า

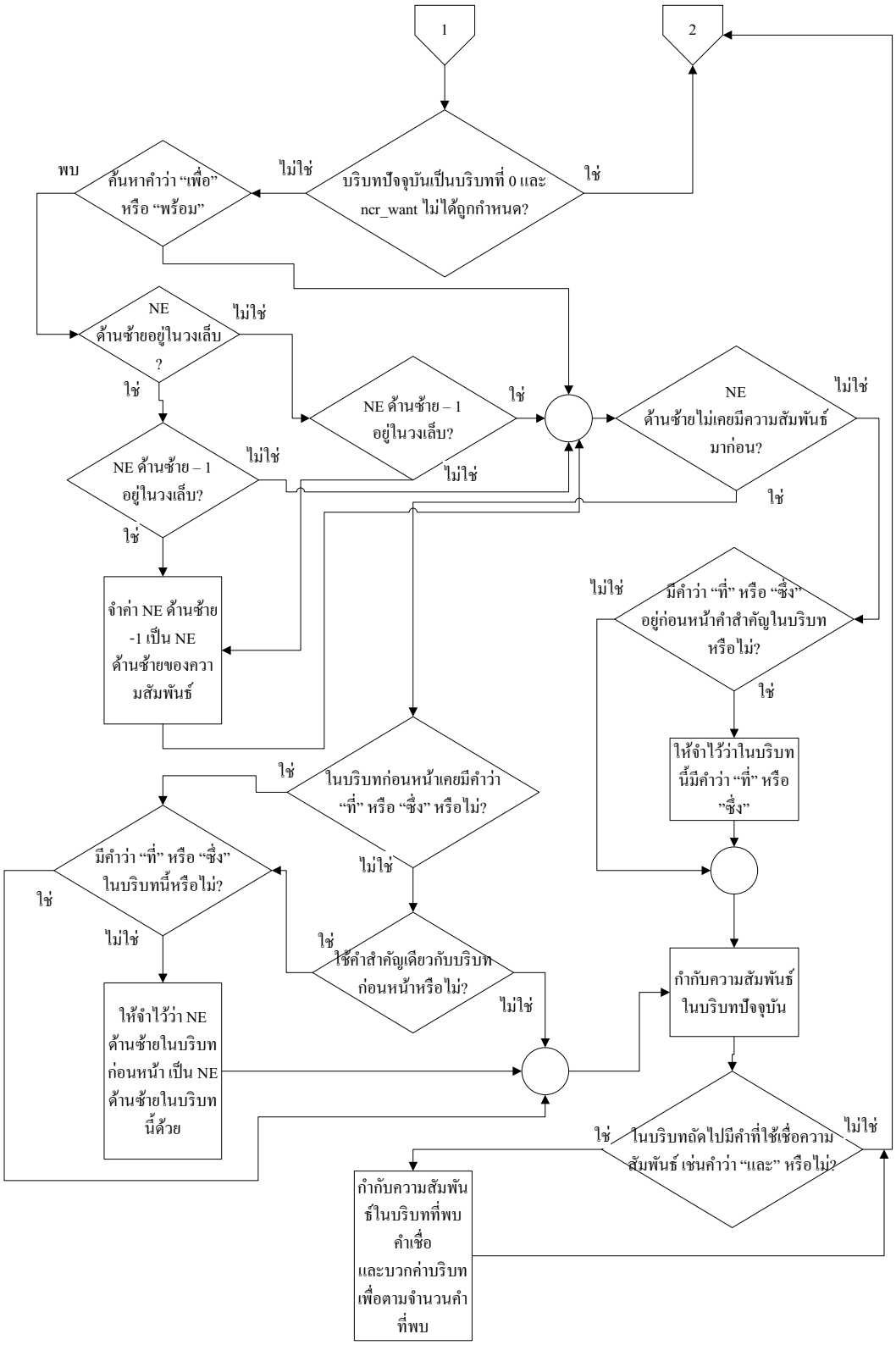
แต่ทั้งนี้ในบริบทที่สองนั้นต้องไม่มีคำว่า “ที่” หรือ “ซึ่ง” เพราะจะทำให้ความหมายเปลี่ยนไปดังในข้อที่ได้กล่าวมาแล้ว

4.3. ขั้นตอนการทำงานในการสกัดความสัมพันธ์

ขั้นตอนการทำงานในการสกัดความสัมพันธ์นั้น มีขั้นตอนในการทำงานดังในภาพที่ 16



ภาพที่ 16 ผังแสดงขั้นตอนการสกัดความสัมพันธ์


1
2
บริบทปัจจุบันเป็นบริบทที่ 0 และ ncr_want ไม่ได้ถูกกำหนด?
ใช่
ไม่ใช่
ค้นหาคำว่า “เพื่อ” หรือ “พร้อม”
พบ
NE ด้านซ้ายอยู่ในวงเล็บ ?
ไม่ใช่
ใช่
NE ด้านซ้าย – 1 อยู่ในวงเล็บ?
ใช่
ไม่ใช่
NE ด้านซ้าย – 1 อยู่ในวงเล็บ?
ไม่ใช่
ใช่
NE ด้านซ้ายไม่เคยมีความสัมพันธ์มาก่อน?
ไม่ใช่
ใช่
จำค่า NE ด้านซ้าย -1 เป็น NE ด้านซ้ายของความสัมพันธ์
มีคำว่า “ที่” หรือ “ซึ่ง” อยู่ก่อนหน้าคำสำคัญในบริบทหรือไม่?
ไม่ใช่
ใช่
ให้จำไว้ว่าในบริบทนี้มีคำว่า “ที่” หรือ ”ซึ่ง”
ในบริบทก่อนหน้าเคยมีคำว่า “ที่” หรือ “ซึ่ง” หรือไม่?
ใช่
ไม่ใช่
มีคำว่า “ที่” หรือ “ซึ่ง” ในบริบทนี้หรือไม่?
ใช่
ไม่ใช่
ใช้คำสำคัญเดียวกับบริบทก่อนหน้าหรือไม่?
ใช่
ไม่ใช่
ให้จำไว้ว่า NE ด้านซ้ายในบริบทก่อนหน้า เป็น NE ด้านซ้ายในบริบทนี้ด้วย
กำกับความสัมพันธ์ในบริบทปัจจุบัน
ในบริบทถัดไปมีคำที่ใช้เชื่อความสัมพันธ์ เช่นคำว่า “และ” หรือไม่?
ใช่
ไม่ใช่
กำกับความสัมพันธ์ในบริบทที่พบคำเชื่อ และบวกค่าบริบทเพื่อตามจำนวนคำที่พบ

ภาพที่ 16 (ต่อ)

จากภาพที่ 16 แสดงขั้นตอนการทำงานในการสกัดความสัมพันธ์ โดยในขั้นตอนแรกคือการรับข้อความที่ผ่านการสกัด NE แล้วเพื่อมาทำการสกัดความสัมพันธ์ ต่อจากนั้นจะเป็นการทำงานแบบวนซ้ำ (Loop) โดยจะตรวจสอบว่าจบบริบทสุดท้ายแล้วหรือไม่ โดยทุกๆ ครั้งที่วนกลับมาที่ตำแหน่งนี้จะทำการเลื่อนบริบทไปยังบริบทถัดไป (จุดนี้ถือเป็นจุดเริ่มต้นของการตรวจสอบ) หากใช่ระบบจะบันทึกข้อความที่ผ่านการสกัดความสัมพันธ์แล้วลงไฟล์ข้อความ และถือเป็นการเสร็จสิ้นการสกัดความสัมพันธ์ แต่หากไม่ใช่จะเป็นการตรวจสอบวงเล็บโดยกำหนดว่าหากบริบทปัจจุบันมีวงเล็บเปิดเกิดขึ้นและยังไม่มีวงเล็บปิด ให้กลับไปทำงานยังจุดเริ่มต้น จากนั้นจะตรวจสอบว่ามี NE ทางด้านซ้ายหรือไม่ หากไม่ใช่ให้ไปตรวจสอบ NE ทางด้านขวา แต่หากใช่ให้ตรวจสอบว่า NE ทางด้านซ้ายอยู่ในวงเล็บและ NE ทางด้านขวาอยู่นอกวงเล็บใช่หรือไม่ (มีวงเล็บปิดอยู่ในบริบทปัจจุบันใช่หรือไม่) หากไม่ใช่ให้ไปตรวจสอบ NE ทางด้านขวา แต่หากใช่ให้รวมบริบทปัจจุบันกับบริบทที่มีวงเล็บเปิดเกิดขึ้นมาก่อนหน้า โดยในการรวมนั้นให้รวมในส่วนของบริบทที่เกิดขึ้นก่อนวงเล็บเปิด และเกิดขึ้นหลังวงเล็บปิดเข้าด้วยกัน (เพราะถือว่าเป็นบริบทเดียวกัน) จากนั้นจะตรวจสอบ NE ทางด้านขวา โดยตรวจสอบว่ามี NE ทางด้านขวาอยู่หรือไม่ หากไม่ใช่ให้กลับไปยังจุดเริ่มต้น แต่หากใช่ให้ทำการค้นหา NE ทางด้านซ้ายที่ไม่ได้อยู่ในวงเล็บ (หากมีวงเล็บเกิดขึ้นก่อนหน้า) และจำตำแหน่ง NE นั้นเป็น NE ทางด้านซ้าย แต่หากไม่มีวงเล็บเกิดขึ้นให้จำ NE ตำแหน่งปัจจุบัน ต่อไปจะตรวจสอบว่าบริบทปัจจุบันนั้นคือบริบทที่มากกว่า 1 หรือไม่ หากไม่ใช่ให้ไปค้นหาประโยคปฏิเสธ แต่หากใช่ให้ตรวจสอบ NE ทางซ้ายว่ามีลักษณะความเป็นเจ้าของหรือไม่ หากใช่ให้จำตำแหน่งของ NE เอาไว้ จากนั้นจะตรวจสอบว่าบริบทนี้เป็นประโยคปฏิเสธหรือไม่ หากใช่ให้กลับไปทำงานที่จุดเริ่มต้น แต่หากไม่ใช่ให้ตรวจสอบว่า NE ด้านขวามีลักษณะความเป็นเจ้าของกับ NE อื่นหรือไม่ หากใช่ให้จำตำแหน่ง NE ที่มีความสัมพันธ์ด้วยเอาไว้ ในส่วนนี้จะถูกใช้ในการตรวจสอบความสัมพันธ์ในคุณสมบัติ want_after_ne2 โดยจะตรวจสอบในบริบทหลังจาก NE ที่มีความสัมพันธ์ในลักษณะความเป็นเจ้าของ (ถ้ามี) หลังจากตรวจสอบ NE ทางขวาที่มีลักษณะความเป็นเจ้าของแล้ว ต่อไปจะเป็นการตรวจสอบความสัมพันธ์ด้วยคำกริยา (ดังเช่นในภาพที่ 2) หากไม่มีความสัมพันธ์เกิดขึ้นจะทำการค้นหาความสัมพันธ์ด้วยคำบุพบท หากยังไม่มีความสัมพันธ์เกิดขึ้น จะกลับไปทำงานยังจุดเริ่มต้น แต่หากพบความสัมพันธ์ขึ้นในขั้นตอนใดขั้นตอนหนึ่ง จะทำการตรวจสอบว่าบริบทปัจจุบันเป็นบริบทที่ 0 และ ncr_want ไม่ได้ถูกกำหนด

ใช่หรือไม่ หากใช่จะหมายความว่า มีความสัมพันธ์เกิดขึ้นในบริบทปัจจุบันแต่ไม่มี NE ทางด้านซ้าย (เพราะเป็นบริบทที่ 0) ดังนั้นจึงให้กลับไปยังจุดเริ่มต้น แต่หากไม่ใช่ให้ค้นหาว่ามีคำว่า “เพื่อ” หรือ “พร้อม” อยู่ในบริบทหรือไม่ โดยเงื่อนไขการค้นหาให้เป็นไปตามกฎที่ได้ตั้งไว้ หากไม่พบให้ไปตรวจสอบว่า NE ด้านซ้ายเคยมีความสัมพันธ์มาก่อนหรือไม่ แต่หากพบคำว่า “เพื่อ” หรือ “พร้อม” อยู่ในบริบท จะทำการตรวจสอบว่าบริบทปัจจุบันอยู่ในวงเล็บหรือไม่ หากใช่หรือไม่ใช่ NE ด้านซ้ายในบริบทก่อนหน้า จะต้องเป็นลักษณะเดียวกัน หากไม่ใช่จะไปทำการตรวจสอบความสัมพันธ์ของ NE ด้านซ้ายที่เคยมีมาก่อน แต่หากใช่ให้จำตำแหน่งของ NE ด้านซ้ายในบริบทก่อนหน้าเป็น NE ด้านซ้ายของบริบทนี้ จากนั้นจะไปตรวจสอบว่า NE ด้านซ้ายของบริบทนี้เคยมีความสัมพันธ์กับ NE อื่นมาก่อนแล้วหรือไม่ หากใช่ให้ข้ามไปตรวจสอบคำว่า “ที่” หรือ “ซึ่ง” ในบริบทก่อนหน้า แต่หากไม่ใช่ให้ตรวจสอบว่าในบริบทปัจจุบันมีคำว่า “ที่” หรือ “ซึ่ง” ก่อนหน้าคำสำคัญหรือไม่ หากไม่ใช่ให้กำกับความสัมพันธ์ในบริบทปัจจุบันตามประเภทของคำสำคัญ แต่หากใช่ให้จำเอาไว้ว่าบริบทนี้มีคำว่า “ที่” หรือ “ซึ่ง” เกิดขึ้นมา ย้อนกลับไปในเงื่อนไขที่ตรวจสอบว่าเคยมีคำว่า “ที่” หรือ “ซึ่ง” เกิดขึ้นมาในบริบทก่อนหน้าหรือไม่ หากใช่ให้ไปกำกับความสัมพันธ์ในบริบทนี้ตามปกติ แต่หากไม่ใช่ให้กำกับความสัมพันธ์ระหว่าง NE ด้านซ้ายในบริบทก่อนหน้ากับ NE ด้านขวาในบริบทปัจจุบัน แต่หากไม่มีคำว่า “ที่” หรือ “ซึ่ง” เกิดขึ้นในบริบทก่อนหน้า ให้ตรวจสอบว่าบริบทนี้ได้ใช้คำสำคัญคำเดียวกับบริบทก่อนหน้าหรือไม่ ซึ่งหากไม่ใช่ก็ให้กำกับความสัมพันธ์ตามปกติในบริบทนี้ แต่หากใช่ให้ตรวจสอบว่ามีคำว่า “ที่” หรือ “ซึ่ง” เกิดขึ้นในบริบทนี้หรือไม่ หากใช่ให้กำกับความสัมพันธ์ตามปกติ แต่หากไม่ใช่ให้กำกับความสัมพันธ์ระหว่าง NE ด้านซ้ายของบริบทก่อนหน้ากับ NE ด้านขวาของบริบทปัจจุบัน หลังจากกำกับความสัมพันธ์ในบริบทปัจจุบันแล้ว จะทำการตรวจสอบว่าในบริบทถัดไปมีคำที่สามารถเชื่อมความสัมพันธ์ เช่นคำว่า “และ”, “หรือ” หรือไม่ หากไม่ใช่ให้กลับไปเริ่มทำงานที่จุดเริ่มต้น แต่หากใช่ให้กำกับความสัมพันธ์ในบริบทที่พบคำเชื่อมในชนิดเดียวกับบริบทแรก และบวกค่าตำแหน่งบริบทเพิ่มขึ้นตามจำนวนบริบทที่พบคำเชื่อม จากนั้นให้กลับไปทำงานที่จุดเริ่มต้นอีกครั้ง

5. ประเมินและสรุปผลการทดลอง

ทำการประเมินผลการทดลองโดยการหาค่า Recall, Precision, และค่า F จากนั้นสรุปผลการวิจัยและจัดทำรายงานวิทยานิพนธ์

บทที่ 5

ผลการวิจัย

จากการทดลองสามารถแบ่งผลการทดลองได้เป็นสองประเภทหลักๆ คือ ส่วนของการสกัด NE และส่วนของการสกัดความสัมพันธ์ โดยในการทดลองของทั้งสองส่วนนั้นจะใช้ชุดข้อมูลที่จะนำมาทดสอบสองชุด คือ ชุดที่ใช้ในการฝึกฝนระบบ (Training Set) จำนวน 100 ข้อความ และชุดที่ใช้ในการทดสอบจริง (Test Set) จำนวน 200 ข้อความ โดยจะนำข้อความเหล่านี้มาทำการการสกัด NE ก่อนและจึงนำผลลัพธ์ที่ได้มาทำการสกัดความสัมพันธ์ต่อไป

สำหรับในส่วนของการสกัด NE และการสกัดความสัมพันธ์นั้นจะต้องทำการทดสอบกับชุดข้อมูลที่ใช้ในการฝึกฝน (100 ข้อความ) ก่อน เพื่อเป็นการฝึกฝนระบบ กล่าวคือ เมื่อทำการทดสอบในชุดฝึกฝนแล้วจะทำการตรวจสอบผลลัพธ์ และปัญหาที่เกิดขึ้น จากนั้นจึงทำการปรับแต่งระบบ คือ การปรับปรุงกฎเดิมที่มีอยู่ (เช่น การปรับระยะห่างของคำสำคัญกับ NE), การสร้างกฎเพิ่มเติมเข้าไปในระบบ (เช่น เพิ่มกฎของความสัมพันธ์ที่มีการกำหนด NE ทางด้านซ้าย) และการแก้ไขข้อผิดพลาดของโปรแกรม (Bug) เพื่อให้ได้ผลลัพธ์ของชุดข้อมูลนี้ให้มีความแม่นยำมากที่สุดเท่าที่จะเป็นไปได้ และจากนั้นจะนำระบบที่ได้ผ่านการปรับแต่งแล้วมาทำการวิเคราะห์ชุดข้อมูลในส่วนที่สอง (200 ข้อความ) และวัดผลการข้อวิเคราะห์ออกมาเป็นค่า Precision, Recall, และค่า F

สำหรับวิธีการวัดผลของค่าต่างๆ มีดังนี้

1. ค่า F

$$F = \frac{(2 \times Precision \times Recall)}{(Precision + Recall)}$$

2. ค่า Precision

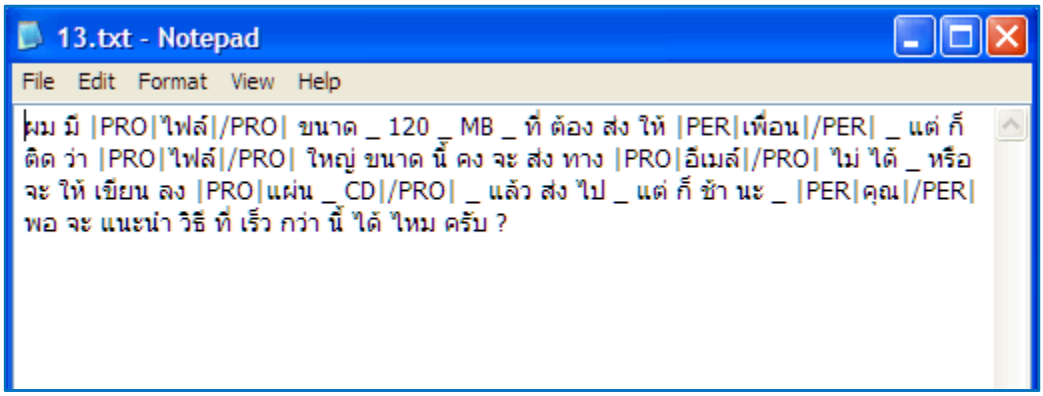
$$Precision = \frac{(\text{จำนวนคำตอบที่ถูกต้องที่ระบบค้นพบ} \times 100)}{\text{จำนวนคำตอบทั้งหมดที่ระบบเลือก}}$$

3. **ค่า Recall**

$$Recall = \frac{(\text{จำนวนคำตอบที่ถูกต้องที่ระบบค้นพบ} \times 100)}{\text{จำนวนคำตอบที่ถูกต้องทั้งหมดในเอกสาร}}$$

ผลการสกัด NE

การสกัด NE ระบบจะทำการรับข้อความที่อยู่ในรูปแบบไฟล์ข้อความ (Text File) เข้าไปประมวลผลและทำการวิเคราะห์ข้อความ จากนั้นจึงกำกับประเภท NE ให้กับคำต่างๆ ในข้อความนั้นที่ระบบตรวจพบ จากนั้นจึงส่งผลลัพธ์ออกมาเป็นข้อความที่ถูกกำกับ NE แล้ว ในรูปแบบของไฟล์ข้อความเช่นเดียวกัน ดังเช่นภาพที่ 17 (สามารถดูตัวอย่างผลลัพธ์เพิ่มเติมได้ที่ภาคผนวก ข หน้า 128-129)



ภาพที่ 17 ผลลัพธ์ของการสกัด NE

สำหรับวิธีการวัดผลจะถูกแบ่งเป็น 2 ส่วน คือ ส่วนของชุดฝึกฝน (Training Set) และส่วนของชุดทดสอบจริง (Test Set) ดังนี้ (สามารถดูผลการสกัด NE ทั้งหมดได้ที่ภาคผนวก ข หน้า 99-112)

1. **ชุดฝึกฝน (Training Set)**

สำหรับในชุดฝึกฝนนี้ ระบบสามารถสกัด NE ได้ถูกต้องจำนวน 833 คำ, สกัดได้แต่ผิดจำนวน 13 คำ, สกัดไม่ได้ 44 คำ

ดังนั้น คำตอบที่ถูกต้องทั้งหมดในข้อความ (ไม่รวมปัญหาตัดคำ) จะได้เท่ากับ 877 คำ (NE ที่สกัดได้ถูกต้อง 833 คำ + NE ที่สกัดไม่ได้ 44 คำ), คำตอบที่ระบบเลือกมาทั้งหมด จะได้

เท่ากับ 846 คำ (NE ที่สกัดได้ถูกต้อง 833 คำ + NE ที่สกัดได้แต่ผิด 13 คำ) ดังนั้นในการหาค่าประสิทธิภาพต่างๆ จะสามารถแทนค่าได้ดังนี้

ค่า Recall

$$Recall = \frac{(833 \times 100)}{877} = 94.98$$

ค่า Precision

$$Precision = \frac{(833 \times 100)}{846} = 98.46$$

ค่า F

$$F = \frac{(2 \times 98.46 \times 94.98)}{(98.46 + 94.98)} = 96.69$$

2. ชุดทดสอบจริง (Test Set)

สำหรับในชุดทดสอบจริง ระบบสามารถสกัด NE ได้ถูกต้องจำนวน 1359 คำ, สกัดได้แต่ผิดจำนวน 21 คำ, สกัดไม่ได้ 171 คำ

ดังนั้น คำตอบที่ถูกต้องทั้งหมดในข้อความ (ไม่รวมปัญหาตัดคำ) จะได้เท่ากับ 1530 คำ (NE ที่สกัดได้ถูกต้อง 1359 คำ + NE ที่สกัดไม่ได้ 171 คำ), คำตอบที่ระบบเลือกมาทั้งหมด จะได้เท่ากับ 1380 คำ (NE ที่สกัดได้ถูกต้อง 1359 คำ + NE ที่สกัดได้แต่ผิด 21 คำ) ดังนั้นในการหาค่าประสิทธิภาพต่างๆ จะสามารถแทนค่าได้ดังนี้

ค่า Recall

$$Recall = \frac{(1359 \times 100)}{1530} = 88.82$$

ค่า Precision

$$Precision = \frac{(1359 \times 100)}{1380} = 98.48$$

ค่า F

$$F = \frac{(2 \times 98.48 \times 88.82)}{(98.48 + 88.82)} = 93.40$$

ผลของการวัดประสิทธิภาพของระบบสกัด NE แสดงอยู่ในตารางที่ 8 ซึ่งจะไม่นับรวมผลที่เกิดจากคำที่สะกดผิดและการตัดคำที่ผิดพลาด

ตารางที่ 8 ผลการวัดประสิทธิภาพการสกัด NE

	Recall (%)	Precision (%)	F-Measure (%)
ชุดฝึกฝนระบบ	94.98	98.46	96.69
ทดสอบจริง	88.82	98.48	93.40

อภิปรายผลและปัญหาของการสกัด NE

วิธีการในการเลือกคำเพื่อนำไปตรวจสอบกับพจนานุกรมรายชื่อด้วยวิธีการผสมคำทีละคำและเลือกคำที่ตรงกับในรายชื่อมากที่สุด ซึ่งเป็นการประยุกต์มาจากวิธีการของ Charoenpornsawat และคณะ (Charoenpornsawat, Kijsirikul and Maknavin 1998) (ซึ่งใช้การกำกับหน้าที่ของคำและนำคำที่ถูกกำกับเป็นชื่อเฉพาะไปทำการค้นหา) ให้ผลลัพธ์ที่ดีในการเลือกรายชื่อที่มีความเป็นไปได้ที่จะเป็น NE ซึ่งสามารถสังเกตได้จากค่า Precision ที่ค่อนข้างสูงทั้งในชุดข้อความฝึกฝนระบบ และชุดทดสอบจริง (98.46% และ 98.48% ตามลำดับ) ซึ่งหมายความว่าเมื่อระบบเลือก NE ออกมาแล้วมีความผิดพลาดที่ต่ำนั่นเอง

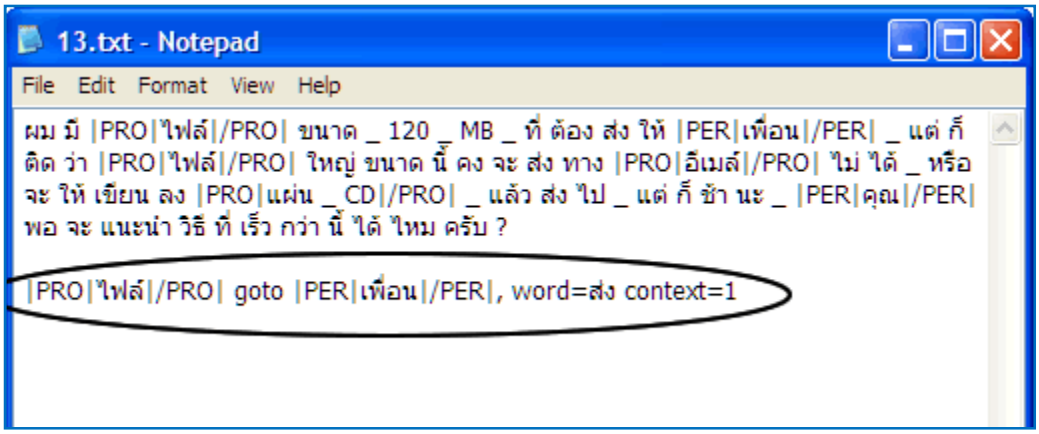
แต่ถึงแม้ค่า Precision ของระบบนี้อยู่ในระดับที่สูง แต่ทั้งนี้ทั้งนั้นระบบก็ยังมีข้อผิดพลาดอยู่บ้างในการดึงคำตอบออกมา เช่น คำบางคำที่มีความหมายได้หลายอย่าง ยกตัวอย่างเช่นคำว่า สามารถ, แพร่, รายงาน เป็นต้น ซึ่งคำเหล่านี้มีความหมายได้หลายอย่าง จึงสามารถทำให้ระบบเกิดการเข้าใจผิดได้ ซึ่งปัญหาเหล่านี้เกิดจากการที่ไม่มีคลังมูลที่เป็นตัวอย่างประโยคแบบต่างๆ ให้ระบบเรียนรู้ลักษณะการเกิดขึ้นของคำต่างๆ เพื่อที่จะช่วยลดการเข้าใจผิดเมื่อพบคำที่มีความหมายกำกวม

จากสาเหตุที่จำนวนรายชื่อที่นำมาใช้ในการสกัด NE ในระบบนี้มีจำนวนไม่มากนัก มีผลให้ไม่สามารถสกัด NE บางคำได้เนื่องจากระบบไม่รู้จักคำเหล่านั้น แต่เนื่องจากมีการใช้กฎเข้ามาช่วยในการสกัด NE จึงทำให้ระบบสามารถสกัด NE บางคำที่ไม่มีอยู่ในพจนานุกรมรายชื่อได้ ยกตัวอย่างเช่นคำว่า “ไฟล์ normal.dot” จากคำนี้ระบบจะรู้จักแต่คำว่า “ไฟล์” แต่สำหรับคำว่า “normal.dot” นั้นไม่ได้มีบันทึกเอาไว้ในพจนานุกรมรายชื่อ ซึ่งหากระบบทำการเปรียบเทียบคำกับพจนานุกรมรายชื่อเพียงอย่างเดียวก็จะกำกับคำว่า “ไฟล์” เพียงคำเดียวที่เป็น NE แต่ในความเป็นจริงคำว่า “normal.dot” คือชื่อไฟล์ จึงสมควรที่จะเป็น NE ร่วมกับคำว่า “ไฟล์” ด้วย แต่ด้วยการที่มี

การสร้างกฎเอาไว้ว่าเมื่อค้นพบคำที่ตรงกับพจนานุกรมรายชื่อแล้วให้ค้นหาคำที่อยู่ถัดไปด้วยว่าเป็นคำภาษาอังกฤษหรือไม่ (อธิบายไว้ในบทที่ 4) จึงทำให้ระบบสามารถที่จะสกัดได้ว่า “ไฟล์ normal.dot” เป็น NE คำเดียวกัน จากกฎที่สร้างขึ้นนี้ทำให้ช่วยลดความผิดพลาดจากการสกัด NE ที่ไม่ได้ถูกบันทึกไว้ในพจนานุกรมรายชื่อได้ในระดับหนึ่ง

ผลการสกัดความสัมพันธ์

ในการสกัดความสัมพันธ์จะนำข้อความที่ได้จากการสกัด NE มาทำการวิเคราะห์ โดยการป้อนข้อความเข้าไปให้แก่ระบบ และเมื่อระบบทำการประมวลผลแล้ว หากพบความสัมพันธ์ในข้อความนั้น จะบอกลักษณะความสัมพันธ์ทั้ง NE ที่มีความสัมพันธ์, คำที่ก่อให้เกิดความสัมพันธ์, ประเภทของความสัมพันธ์, และตำแหน่งของบริบทที่มีความสัมพันธ์ไว้ที่ใต้ข้อความต้นฉบับ ดังเช่นภาพที่ 18 (สามารถดูตัวอย่างผลลัพธ์เพิ่มเติมได้ที่ภาคผนวก ข หน้า 130-132)



ภาพที่ 18 ผลลัพธ์ของการสกัดความสัมพันธ์ (ในวงรี)

สำหรับวิธีการวัดผลจะถูกแบ่งเป็นสองส่วน คือ ชุดฝึกฝน (Training Set) และชุดทดสอบจริง (Test Set) ดังนี้ (สามารถดูผลการสกัดความสัมพันธ์ทั้งหมดได้ที่ภาคผนวก ข หน้า 113-127)

1. **ชุดฝึกฝน (Training Set)**

สำหรับในชุดฝึกฝน สามารถสกัดความสัมพันธ์ได้ถูกต้อง จำนวน 172 ความสัมพันธ์, สกัดได้แต่ผิด จำนวน 10 ความสัมพันธ์, และสกัดไม่ได้ จำนวน 23 ความสัมพันธ์

ดังนั้น คำตอบที่ถูกต้องทั้งหมดในข้อความ (ไม่รวมปัญหาการตัดคำ, การสกัด NE, และการสะกดผิด) จะได้เท่ากับ 195 ความสัมพันธ์ (ความสัมพันธ์ที่สกัดได้ถูกต้อง 172 ความสัมพันธ์ + ความสัมพันธ์ที่สกัดไม่ได้ 23 ความสัมพันธ์), และคำตอบที่ระบบเลือกมาทั้งหมด จะได้เท่ากับ 182 ความสัมพันธ์ (ความสัมพันธ์ที่สกัดได้ถูกต้อง 172 ความสัมพันธ์ + ความสัมพันธ์ที่สกัดได้แต่ผิด 10 ความสัมพันธ์) ดังนั้นในการหาค่าประสิทธิภาพต่างๆ จะสามารถแทนค่าได้ดังนี้

ค่า Recall

$$Recall = \frac{(172 \times 100)}{195} = 88.21$$

ค่า Precision

$$Precision = \frac{(175 \times 100)}{182} = 94.51$$

ค่า F

$$F = \frac{(2 \times 94.51 \times 88.21)}{(94.51 + 88.21)} = 91.25$$

2. ชุดทดสอบจริง (Test Set)

สำหรับในส่วนของการทดสอบจริง สามารถสกัดความสัมพันธ์ได้ถูกต้อง จำนวน 244 ความสัมพันธ์, สกัดได้แต่ผิด จำนวน 30 ความสัมพันธ์, และสกัดไม่ได้ จำนวน 56 ความสัมพันธ์

ดังนั้น คำตอบที่ถูกต้องทั้งหมดในข้อความ (ไม่รวมปัญหาการตัดคำ, การสกัด NE, และการสะกดผิด) จะได้เท่ากับ 300 ความสัมพันธ์ (ความสัมพันธ์ที่สกัดได้ถูกต้อง 244 ความสัมพันธ์ + ความสัมพันธ์ที่สกัดไม่ได้ 56 ความสัมพันธ์), และคำตอบที่ระบบเลือกมาทั้งหมด จะได้เท่ากับ 274 ความสัมพันธ์ (ความสัมพันธ์ที่สกัดได้ถูกต้อง 244 ความสัมพันธ์ + ความสัมพันธ์ที่สกัดได้แต่ผิด 30 ความสัมพันธ์) ดังนั้นในการหาค่าประสิทธิภาพต่างๆ จะสามารถแทนค่าได้ดังนี้

ค่า Recall

$$Recall = \frac{(244 \times 100)}{300} = 81.33$$

ค่า Precision

$$Precision = \frac{(244 \times 100)}{274} = 89.05$$

ค่า F

$$F = \frac{(2 \times 89.05 \times 81.33)}{(89.05 + 81.33)} = 85.02$$

ผลการวัดประสิทธิภาพของการสกัดความสัมพันธ์แสดงอยู่ในตารางที่ 9 ซึ่งผลของการสกัดความสัมพันธ์ที่วัดได้นี้จะไม่นับรวมความผิดพลาดที่เกิดจาก การตัดคำที่ผิดพลาด, การสกัด NE ที่ผิดพลาด, และการสะกด

ตารางที่ 9 ผลการวัดประสิทธิภาพการสกัดความสัมพันธ์

	Recall (%)	**Precision (%)**	**F-Measure (%)**
ชุดฝึกฝนระบบ	88.21	94.51	91.25
ทดสอบจริง	81.33	89.05	85.02

อภิปรายผลและปัญหาของการสกัดความสัมพันธ์

ค่าระลึก (Recall)

จากค่าระลึก (Recall) ในการทดสอบจริง (81.33%) แสดงให้เห็นว่าในการสกัดความสัมพันธ์นั้นยังมีความผิดพลาดที่ทำให้ไม่สามารถสกัดความความสัมพันธ์ออกมาได้ ซึ่งสาเหตุที่ทำให้ไม่สามารถสกัดความสัมพันธ์ออกมาได้นั้นเป็นเพราะว่าในบางครั้งความสัมพันธ์ของ NE ในข้อความไม่ได้เกิดขึ้นโดยมีลักษณะที่ตายตัว และไม่สามารถที่จะกำหนดรูปแบบหรือกฎใดๆ (ซึ่งเป็นวิธีการที่ใช้ในระบบนี้) ให้สามารถที่จะสกัดความสัมพันธ์ในลักษณะนี้ได้ และปัญหาดังกล่าวซึ่งได้ค้นพบจากการพัฒนาระบบสกัดความสัมพันธ์นี้ มีรูปแบบต่างๆ ดังนี้

รูปแบบที่ 1

การกล่าวถึงความสัมพันธ์ในลักษณะที่ผู้อ่านข้อความจำเป็นจะต้องเข้าใจความหมายหรือลักษณะของ NE ยกตัวอย่างเช่น

ตัวอย่างที่ 1: |PRO|SpywareBlaster|/PRO| _ จะ ตรวจ จับ _ |PRO|ActiveX _ controller|/PRO| _ ของ |PRO|สปายแวร์|/PRO|

จากประโยคตัวอย่างให้สังเกตในส่วนที่ขีดเส้นใต้ หากเราอ่านประโยคนี้จะสามารถเข้าใจได้ว่า |PRO|ActiveX _ controller|/PRO| เป็นส่วนหนึ่งของ |PRO|สปายแวร์|/PRO| หรืออีกนัยหนึ่งก็คือ |PRO|ActiveX _ controller|/PRO| ติดตั้งอยู่ใน |PRO|สปายแวร์|/PRO| (located_in) นั่นเอง ซึ่งถ้าหากพิจารณาการใช้คำในประโยคแล้วจะพบคำที่สื่อถึงความสัมพันธ์ คือคำว่า “ของ” ซึ่งทำให้เราสามารถเข้าใจได้ว่า |PRO|ActiveX _ controller|/PRO| ติดตั้งอยู่ใน |PRO|สปายแวร์|/PRO| ซึ่งไม่น่าจะเกิดปัญหาอะไรหากนำคำนี้มารวมอยู่ในรายชื่อคำสำคัญที่สื่อถึงความสัมพันธ์ หากแต่ว่าไม่สามารถทำเช่นนั้นได้เนื่องมาจากคำว่า “ของ” เมื่อนำไปใช้กับ NE อื่นอาจจะทำให้ความสัมพันธ์ที่สื่อออกมามีความหมายต่างออกไปได้ซึ่งขึ้นอยู่กับประเภทของ NE ยกตัวอย่างเช่น

ตัวอย่างที่ 2: |PRO|กล้องดิจิตอล|/PRO| ของ |ORG|ฟูจิ|/ORG|

จากประโยคนี้จะเห็นว่ามีโครงสร้างเหมือนกับประโยคก่อนหน้า คือมีคำว่า “ของ” ที่เป็นคำที่สื่อถึงความสัมพันธ์เช่นเดียวกัน แต่หากเราพิจารณาจากความหมายของประโยคจริงๆ แล้ว จะพบว่าความหมายที่สื่อออกมานั้นไม่เหมือนกัน ซึ่งในประโยคนี้มีความหมายว่า |PRO|กล้องดิจิตอล|/PRO| เป็นผลิตภัณฑ์ของ |ORG|ฟูจิ|/ORG| หรืออีกนัยหนึ่งก็คือ |ORG|ฟูจิ|/ORG| ผลิต |PRO|กล้องดิจิตอล|/PRO| (create) นั่นเอง

จากทั้งสองตัวอย่างที่กล่าวมา ทำให้ทราบว่าคำบางคำนั้นสามารถใช้บ่งบอกถึงความสัมพันธ์ได้ แต่ลักษณะของความสัมพันธ์นั้นจะสามารถเปลี่ยนแปลงไปได้ตามแต่ประเภท NE ที่คำๆ นั้นเกี่ยวข้อง ซึ่งสิ่งที่จะบอกว่าเป็นความสัมพันธ์อะไรนั้น คือความรู้และประสบการณ์ของผู้อ่านข้อความเอง ดังเช่นในตัวอย่างแรก ผู้อ่านจำเป็นต้องที่จะต้องรู้จักด้วยว่า “ActiveX _ controller” และ “สปายแวร์” คืออะไร มีลักษณะเป็นอย่างไร จึงจะสามารถเข้าใจได้ว่าคำว่า “ของ” ในที่นี้หมายถึง “ActiveX _ controller เป็นส่วนหนึ่งของ สปายแวร์” เช่นเดียวกับตัวอย่างที่สอง ผู้อ่านจำเป็นต้องมีความรู้มาก่อนว่า “ฟูจิ” นั้นเป็นบริษัทที่ทำธุรกิจเกี่ยวกับอะไร และ “กล้องดิจิตอล” คืออะไร จึงจะสามารถเข้าใจได้ว่าคำว่า “ของ” ในที่นี้หมายถึง “กล้องดิจิตอล เป็นผลิตภัณฑ์ของ ฟูจิ”

ตัวอย่างที่ 3: เข้า ไป ยัง |PRO|อินเทอร์เน็ต|/PRO| จาก สถานที่ แตกต่าง กัน _ เพื่อเข้าถึง |PRO|คอมพิวเตอร์|/PRO| ของ |ORG|บริษัท|/ORG|

ตัวอย่างที่ 3 คืออีกหนึ่งตัวอย่างของความสัมพันธ์ลักษณะนี้ ให้สังเกตว่าตัวอย่างนี้มีความคล้ายคลึงกับในตัวอย่างที่ 2 เป็นอย่างมาก (ในส่วนที่ขีดเส้นใต้) คือ NE แรกเป็นชนิด PRO และ NE ที่สองเป็น ORG แต่เมื่อพิจารณาถึงความหมายในข้อความแล้ว จะเห็นว่ามีความหมายที่ต่างกัน คือ |PRO|คอมพิวเตอร์|/PRO| เป็นทรัพย์สินของ |ORG|บริษัท|/ORG| หรืออีกนัยหนึ่งก็คือ |PRO|คอมพิวเตอร์|/PRO| อยู่ใน |ORG|บริษัท|/ORG| (located_in) นั่นเอง ซึ่งในประโยคนี้ก็จำเป็นที่จะต้องอาศัยความรู้ของผู้อ่านเช่นเดียวกันในการที่จะพิจารณาว่าควรจะมีความสัมพันธ์ในลักษณะใด

รูปแบบที่ 2

การเข้าใจความหมายของคำภายในบริบท ซึ่งจะสามารถทำให้ผู้อ่านเข้าใจความหมายของข้อความได้มากกว่าการมองหาคำสำคัญเพียงไม่กี่คำ ซึ่งสิ่งที่แตกต่างจากลักษณะอื่นๆ ก็คือ ความหมายในบริบทนี้มีส่วนสำคัญต่อความหมายของข้อความ ซึ่งหากไม่เข้าใจความหมายในส่วนนี้แล้วอาจจะทำให้เข้าใจความหมายของข้อความผิดเพี้ยนไปได้

ตัวอย่างที่ 1: |PER|คุณ|/PER| จะ ต้อง เตรียม |PRO|กระดาษ พิมพ์|/PRO| นามบัตร ขนาด _ 2 _ x _ 3.5 _ นิ้ว _ ซึ่ง อาจ จะ ใช้ |PRO|กระดาษ _ A4|/PRO| _ ที่ แข็ง สักหน่อย ก็ ได้ _ จากนั้น เข้า ไป ที่ |PRO|เว็บไซต์ _ Microsoft|/PRO| _ เพื่อ ดาวน์โหลด เทมเพลต

จากตัวอย่าง ให้สังเกตในส่วนที่ขีดเส้นใต้ หากให้ระบบทำการสกัดความสัมพันธ์ของข้อความนี้ ในส่วนที่ขีดเส้นใต้ระบบจะทำการสกัดความสัมพันธ์ออกมาว่า “|PRO|กระดาษ _ A4|/PRO| goto |PRO|เว็บไซต์ _ Microsoft|/PRO|” เนื่องจากในบริบทระหว่าง NE ทั้งสอง มีคำว่า “เข้าไปที่” จึงทำให้ระบบเข้าใจว่า “|PRO|กระดาษ _ A4|/PRO| เข้าไปที่ |PRO|เว็บไซต์ _ Microsoft|/PRO|” แต่ในความเป็นจริงหากเราอ่านข้อความแล้ว จากประสบการณ์ของผู้อ่านที่เป็นมนุษย์จะทราบว่า “กระดาษ A4” ไม่สามารถที่จะเดินทางเข้าไปที่ “เว็บไซต์ Microsoft” ได้ (รูปแบบที่ 1) จึงทำให้ทราบว่าไม่สามารถที่จะเกิดความสัมพันธ์แบบ “goto” ได้ ถึงแม้ในบริบทจะมีคำว่า “เข้าไปที่” ก็ตาม แต่ทั้งนี้ผู้อ่านจะทราบได้อย่างไรว่าคำว่า “เข้าไปที่” ในข้อความนั้นกำลังหมายถึงสิ่งใด “เข้าไปที่เว็บไซต์ Microsoft”

หากเราอ่านข้อความในบริบทดังกล่าว จะพบว่ามีคำว่า “จากนั้น” ซึ่งหมายความว่า ข้อความกำลังบอกให้บางสิ่งบางอย่างกระทำขั้นตอนต่อไป ซึ่งก็คือการ “เข้าไปที่เว็บไซต์ Microsoft” นั่นหมายความว่าจะต้องมีการกระทำที่เกิดขึ้นมาก่อนหน้าบริบทนี้ และในการกระทำนั้นจะต้องมี NE ที่ผู้เขียนข้อความต้องการจะบอกให้ “เข้าไปที่เว็บไซต์ Microsoft” ดังนั้นหากเราเริ่มอ่านข้อความตั้งแต่ต้นข้อความจะพบว่ามีการกระทำที่เกิดขึ้นมาก่อนหน้าคือ “|PER|คุณ|/PER| จะ ต้อง เตรียม |PRO|กระดาษ พิมพ์|/PRO|” และหากอ่านเรื่อยมาจนถึงบริบทที่เกิดปัญหา จะพบว่า จากความรู้และประสบการณ์ของผู้อ่านจะทำให้ทราบว่ามี NE เพียงชนิดเดียวที่มีความเป็นไปได้ที่จะ “เข้าไปที่เว็บไซต์ Microsoft” ซึ่งก็คือ “|PER|คุณ|/PER|” ดังนั้นความสัมพันธ์ที่ถูกต้องที่จะเกิดขึ้นจะต้องเป็น “|PER|คุณ|/PER| goto |PRO|เว็บไซต์ _ Microsoft|/PRO|”

ตัวอย่างที่ 2: |ORG|บริษัท _ Samsung _ Electronics|/ORG| _ ใน |LOC|เกาหลีใต้|/LOC| _ คือ _ ผู้ ผลิต |PRO|โทรศัพท์มือถือ _ CDMA|/PRO| _ ที่ ใหญ่ ที่สุด ใน |LOC|โลก|/LOC| _ และ ส่วนใหญ่ จะ ผลิต |PRO|มือถือ|/PRO| รุ่น ที่ พับ ได้

จากตัวอย่างที่ 2 ให้สังเกตในส่วนที่ขีดเส้นใต้ เป็นอีกหนึ่งตัวอย่างของความสัมพันธ์ในลักษณะนี้ ซึ่งคำในบริบทที่บ่งบอกว่า NE ทางซ้ายที่มีความสัมพันธ์กับบริบทนี้ไม่ใช่ “|LOC|โลก|/LOC| ก็คือคำว่า “และ” และ NE ที่จะมามีความสัมพันธ์กับประโยคนี้ก็จำเป็นที่จะต้องใช้ความรู้ความเข้าใจของผู้อ่านข้อความจึงจะทราบว่า “|ORG|บริษัท _ Samsung _ Electronics|/ORG|” คือ NE ที่มีความสัมพันธ์ในบริบทนี้

ตัวอย่างที่ 3: ใน |PRO|ประกาศ|/PRO| แจ้ง เตือน ยัง มี |PRO|ช่อง โหว่|/PRO| อื่นๆ _ ที่ _ สำคัญ _ และ พบ ใน แอพพลิเคชัน _ |PRO|Office|/PRO|

จากตัวอย่างที่ 3 ให้สังเกตในส่วนที่ขีดเส้นใต้ ซึ่งเป็นประโยคที่ระบบจะเข้าใจผิด เนื่องจากในบริบทก่อนหน้านั้นมีความสัมพันธ์เกิดขึ้น (|PRO|ช่อง โหว่|/PRO| located_in |PRO|ประกาศ|/PRO|) และในบริบทนี้มีคำว่า “และ” ซึ่งจะทำให้ NE ทางขวาในบริบทนี้ถูกนำเข้าไปมีความสัมพันธ์กับ NE ทางขวาในบริบทก่อนหน้าด้วย (อธิบายไว้ในบทที่ 4) แต่หากเราอ่านข้อความจะพบว่า NE ด้านซ้ายที่เกิดขึ้นกับความสัมพันธ์ดังกล่าวควรจะเป็น “|PRO|ช่อง

โหว่|/PRO|" เพราะหากเราลองพิจารณาจากบริบทจะพบว่าประกอบด้วยข้อความสองส่วนคือ "อื่นๆ _ ที่ _ สำคัญ" และ "พบ ใน แอพพลิเคชัน _" สำหรับในส่วนแรกนั้นใช้ในการขยายความหมายของคำว่า "ช่องโหว่" และในส่วนที่สองนั้นเป็นส่วนที่บอกถึงความสัมพันธ์กับคำว่า "แอพพลิเคชัน office" และคำว่า "และ" นั้นถูกใช้ในการเชื่อมบริบททั้งสองเข้าด้วยกัน ดังนั้นคำว่า "และ" ในที่นี้จึงไม่ได้มีความหมายในแง่ของการรวม NE เข้ามามีความสัมพันธ์กับความสัมพันธ์ก่อนหน้า (ซึ่งระบบเข้าใจว่าเป็นเช่นนั้น) ด้วยเหตุนี้ในข้อความที่มีลักษณะเช่นนี้ผู้อ่านจึงจำเป็นที่จะต้องมีความเข้าใจในความหมายของบริบทต่างๆ ภายในข้อความ เพื่อที่จะได้สามารถจำแนกได้ว่าคำสำคัญบางคำที่เกิดขึ้นในข้อความนั้นมีความหมายว่าอย่างไร

ค่าความแม่นยำ (Precision)

ค่าความแม่นยำในการทดสอบจริงของระบบนี้ (89.05%) หมายความว่าเมื่อระบบได้ทำการเลือกคำตอบขึ้นมาแล้วมีความผิดพลาดของคำตอบไม่มากนัก แต่อย่างไรก็ตามในการเลือกคำตอบของระบบก็ยังพบความผิดพลาดอยู่บ้าง ซึ่งจะเป็นในลักษณะที่ระบบได้พบความสัมพันธ์ในข้อความแต่ในความเป็นจริงแล้วไม่ได้มีความสัมพันธ์นั้นเกิดขึ้นจริง โดยปัญหาในส่วนนี้นั้นก็มีรูปแบบการเกิดในลักษณะเดียวกันกับในส่วนของค่าระลึก (Recall) เช่นเดียวกัน แต่ผลลัพธ์การเกิดในส่วนนี้ทำให้ระบบเข้าใจผิดว่ามีความสัมพันธ์เกิดขึ้น แต่หากในความเป็นจริงกลับไม่ได้มีความสัมพันธ์ใดๆ เกิดขึ้น

รูปแบบที่ 1

รูปนี้มีลักษณะเหมือนกับ รูปแบบที่ 1 ที่เกิดในส่วนของค่าระลึก คือเกิดจากปัญหาในความไม่เข้าใจความหมายหรือลักษณะของ NE ของระบบเช่นเดียวกัน แต่ในที่นี้ปัญหานี้ทำให้เกิดความเข้าใจผิดให้แก่ระบบ ซึ่งทำให้ระบบเข้าใจว่ามีความสัมพันธ์เกิดขึ้น แต่หากในความเป็นจริงแล้วไม่ได้มีความสัมพันธ์นั้นๆ เกิดขึ้นแต่อย่างใด

ตัวอย่าง : เมื่อใด ก็ ตามที่ |PER|คุณ|/PER| เผลอ ติด |PRO|สปายแวร์|/PRO| เข้า ไป

จากตัวอย่าง ในบริบทระหว่าง NE ทั้งสองมีคำสำคัญปรากฏอยู่ คือคำว่า "ติด" ซึ่งเป็นคำประเภท "located_in" จากการที่ค้นพบคำสำคัญนี้ มีผลทำให้ระบบเข้าใจว่ามีความสัมพันธ์เกิดขึ้นกับ NE ทั้งสอง และทำการกำกับความสัมพันธ์ระหว่าง NE ทั้งสองเป็น

ประเภท “located_in” แต่ในความหมายจริงของข้อความเมื่อเราอ่านข้อความเราจะสามารถทราบได้ว่าความสัมพันธ์นั้นไม่สามารถเกิดขึ้นได้ เนื่องจากหากเราเข้าใจความหมายของ NE ทั้งสองแล้วจะทราบว่า |PER|คุณ|/PER| ไม่สามารถมี |PRO|สปายแวร์|/PRO| ติดตั้งอยู่ในตัวได้ เนื่องจาก “คุณ” ในข้อความนี้หมายถึงมนุษย์ซึ่งเป็นผู้อ่านข้อความ ส่วน |PRO|สปายแวร์|/PRO| นั้นคือซอฟต์แวร์ ซึ่งไม่มีความเป็นไปได้ที่จะติดตั้งอยู่ในตัวมนุษย์ ดังนั้นความหมายจริงๆ ที่ผู้เขียนข้อความนี้ต้องการจะสื่อออกมานั้น น่าจะหมายถึงเครื่องคอมพิวเตอร์ (ของ “คุณ”) มากกว่า แต่เหตุที่ในที่นี้ผู้เขียนใช้คำว่า “คุณ” เพียงอย่างเดียวเนื่องมาจากการใช้คำว่า “คุณ” ในที่นี้ จากประสบการณ์ของผู้อ่านจะสามารถเข้าใจได้เองว่าหมายถึงเครื่องคอมพิวเตอร์ ด้วยเหตุนี้การที่ระบบไม่สามารถเข้าใจความหมายหรือลักษณะของ NE หรือความหมายที่แฝงอยู่ในข้อความได้ จึงเป็นเหตุให้ระบบสกัดความสัมพันธ์ผิดได้

รูปแบบที่ 2

รูปแบบนี้ก็มีลักษณะเดียวกันกับรูปแบบที่ 2 ในส่วนของค่าระลึกเช่นเดียวกัน คือข้อความภายในบริบทนั้นมีส่วนสำคัญต่อความหมายของข้อความ

ตัวอย่าง: |PER|เพื่อน|/PER| ผม เคย บอก ว่า _ ใน _ |PRO|Windows _ XP|/PRO| _ มี |PRO|โปรแกรม|/PRO| ที่ สามารถ ออกเสียงได้

จากตัวอย่างในส่วนที่ขีดเส้นใต้ ในบริบทจะพบคำสำคัญ คือคำว่า “ใน” โดยหากไม่เข้าใจความหมายของข้อความอื่นในบริบทแล้ว จะทำให้เข้าใจว่ามีความสัมพันธ์ “|PER|เพื่อน|/PER| located_in |PRO|Windows _ XP|/PRO|” ซึ่งไม่ถูกต้อง แต่หากเข้าใจความหมายของบริบทดังกล่าวแล้ว จะพบว่ามีสองส่วนที่สำคัญ คือ “เคย บอก ว่า” และ “ใน” ซึ่งในส่วนแรกนั้นใช้เป็นเครื่องบอกว่า “|PER|เพื่อน|/PER| ได้บอกบางสิ่งบางอย่าง” และในส่วนของคำว่า “ใน” นั้นคือ “สิ่งที่ |PER|เพื่อน|/PER| ได้บอกออกมา” ซึ่งไม่ใช่คำที่บอกว่า “|PER|เพื่อน|/PER| อยู่ใน |PRO|Windows _ XP|/PRO|” แต่อย่างใด ดังนั้นจากการที่ไม่เข้าใจความหมายของบริบท จึงทำให้เกิดความเข้าใจผิด และเป็นเหตุให้ระบบเข้าใจผิดว่ามีความสัมพันธ์เกิดขึ้น ทั้งที่ในความเป็นจริงไม่ได้มีความสัมพันธ์เกิดขึ้นแต่อย่างใด

บทที่ 6

สรุปผลการวิจัยและข้อเสนอแนะ

สรุป

วิทยานิพนธ์นี้มีจุดมุ่งหมายเพื่อต้องการจะศึกษาถึงวิธีการที่จะสกัดความสัมพันธ์จากข้อความ เพื่อประโยชน์ในการพัฒนาสาขาวิชาภาษาธรรมชาติ ให้มีความสามารถในการเข้าใจภาษามนุษย์มากขึ้น โดยเฉพาะในภาษาไทยที่การศึกษาในด้านนี้ยังมีไม่มากนัก ดังจะเห็นได้จากเมื่อครั้งที่ผู้ทำวิจัยได้เริ่มต้นทำการวิจัย ยังไม่มีงานวิจัยใดที่ศึกษาในเรื่องความสัมพันธ์ในภาษาไทย ซึ่งจะเป็นประโยชน์อย่างมากในกระบวนการคอมพิวเตอร์ต่างๆ ที่มีการใช้ภาษาเข้ามาเกี่ยวข้อง เช่น ระบบค้นคืนสาระสนเทศต่างๆ (Information Retrieval) ซึ่งจะสามารถช่วยให้ระบบการค้นนั้นสามารถค้นหาได้ตรงตามความต้องการของผู้ใช้ได้มากขึ้น

สำหรับการสกัดความสัมพันธ์ในวิทยานิพนธ์นี้ จะเป็นการสกัดความสัมพันธ์ระหว่างนิพจน์ระบุนามในภาษาไทย ซึ่งนิพจน์ระบุนาม หรือ NE (Named Entity) ก็คือ คำที่ใช้ระบุชื่อเฉพาะชนิดต่างๆ เช่น ชื่อบุคคล ชื่อองค์กร, ชื่อสถานที่, สิ่งที่บ่งบอกถึงวัน เวลา, จำนวนเงิน เป็นต้น โดยที่ประเภท NE ที่ต้องการหาในการวิจัยครั้งนี้มีอยู่ 4 ชนิดคือ ชื่อองค์กร (ORG), ชื่อสถานที่ (LOC), ชื่อบุคคล (PER), และชื่อสิ่งของ (PRO) สำหรับในส่วนของความสัมพันธ์ที่ต้องการสกัดออกมานั้นจะอยู่ระหว่าง NE ทั้งหลายที่ปรากฏอยู่ในข้อความต่างๆ โดยความสัมพันธ์ที่สกัดได้นั้นจะบ่งบอกว่า NE ใดมีความสัมพันธ์กับ NE ใด ในประเภทความสัมพันธ์ชนิดใด ซึ่งชนิดความสัมพันธ์ที่ต้องหาในการวิจัยครั้งมีอยู่ 3 ชนิดคือ ชนิดที่บอกถึงการเดินทาง (goto), ชนิดที่บอกถึงการที่สิ่งใดสิ่งหนึ่งถูกติดตั้งอยู่ในสิ่งใดอีกสิ่งหนึ่ง (located_in), และชนิดที่บอกถึงการสร้างหรือประดิษฐ์ (create)

ขั้นตอนในการวิจัยครั้งนี้แบ่งเป็น 2 ส่วนหลักๆ คือ การสกัด NE และการสกัดความสัมพันธ์ ในส่วนของการสกัด NE นั้นใช้วิธีการนำข้อมูลจากภายนอกเข้ามาช่วยในการ

พิจารณา โดยการใช้พจนานุกรมรายชื่อ และวิธีเลือกคำที่จะนำไปทำการค้นหาในรายชื่อ ที่ได้ประยุกต์มาจากวิธีการของ Charoenpornsawat และคณะ (Charoenpornsawat, Kijsirikul and Maknavin 1998) นั้นก็ได้ให้ผลลัพธ์ที่ดี โดยการตรวจสอบคำเป้าหมายแต่ละคำจากในพจนานุกรมรายชื่อ หากพบคำที่ตรงกับคำในพจนานุกรมรายชื่อมากที่สุดก็ให้เลือกคำนั้นมาเป็น NE นอกจากนั้นยังมีการเพิ่มกฎเข้าไปช่วยในการสกัด NE เพื่อช่วยเพิ่มประสิทธิภาพ (เนื่องจากจำนวนรายชื่อที่นำมาใช้ในการสกัด NE นั้นมีจำนวนที่ไม่มากนัก จึงอาจจะทำให้ประสิทธิภาพของการสกัด NE ไม่ดีเท่าที่ควร) คือ การนำ NE ที่สกัดได้มารวมกับคำภาษาอังกฤษที่มีตำแหน่งอยู่ติดกัน, การค้นหาคำนำหน้าชื่อก่อนที่จะสกัด NE ที่เป็นชื่อบุคคล, การรวม NE สองคำที่มีตำแหน่งติดกันให้เป็น NE เพียงคำเดียว

ต่อมาคือขั้นตอนในการสกัดความสัมพันธ์ โดยใช้กฎฮิวริสติก เข้ามาช่วยในการสกัดความสัมพันธ์ และการค้นหาแบบ Regular Expression ในการค้นหาคำสำคัญต่างๆ ที่อยู่ในข้อความเพื่อที่จะบ่งบอกความสัมพันธ์ โดยที่คำสำคัญต่างๆ จะถูกสร้างขึ้นและมีแบบแผน (Pattern) ในการตรวจสอบต่างๆ กันไป โดยคำสำคัญที่นำมาใช้ในการสกัดความสัมพันธ์นี้จะแบ่งเป็นประเภทคำกริยา (Verb) นอกจากนี้ยังมีคำประเภทอื่นๆ ที่ใช้ในการช่วยตัดสินใจในการสกัดความสัมพันธ์ คือ คำที่บ่งบอกถึงประโยคปฏิเสธ, คำที่บ่งบอกถึงความเป็นเจ้าของ, คำที่บ่งบอกถึงความต่อเนื่องของความสัมพันธ์ เช่น “และ”, “หรือ” เป็นต้น นอกจากการใช้คำสำคัญแล้ว ยังมีการสร้างกฎขึ้นมาเพื่อช่วยในการสกัดความสัมพันธ์ในข้อความที่ลักษณะแบบต่างๆ อีกด้วย เนื่องจากข้อความในภาษามนุษย์นั้นมีรูปแบบในการเขียนได้หลากหลาย จึงจำเป็นที่จะต้องเข้าใจถึงวิธีการใช้ภาษาในข้อความลักษณะต่างๆ ด้วย

ในส่วนของการทดลอง จะใช้โปรแกรมที่เขียนขึ้นมาในการสกัด NE และความสัมพันธ์ โดยจะใช้ข้อความประเภทข่าวสารหรือบทความทางด้านเทคโนโลยีในวงการคอมพิวเตอร์ในช่วงปีตั้งแต่ 2546-2550 จำนวน 300 ข้อความ โดยจะแบ่งเป็น 2 ส่วนคือ ส่วนที่นำมาใช้ในการฝึกฝนระบบ 100 ข้อความ และส่วนทดสอบจริงจำนวน 200 ข้อความ สำหรับในขั้นตอนการทดลองนั้น ข้อความที่นำมาทดสอบจะต้องนำมาผ่านกระบวนการตัดคำ จากนั้นจึงนำไปผ่านกระบวนการสกัด NE เมื่อได้ข้อความที่ผ่านการกำกับ NE เรียบร้อยแล้ว จึงนำข้อความเหล่านั้นมาทำการสกัด

ความสัมพันธ์ จากนั้นจึงวัดประสิทธิภาพในการสกัดทั้ง NE และความสัมพันธ์ออกมาเป็นค่า Recall, Precision, และค่า F

สำหรับประสิทธิภาพในการสกัด NE นั้น ในชุดข้อความฝึกฝนระบบนั้นมีค่า Recall, Precision, และค่า F อยู่ที่ 94.98, 98.46, และ 96.69 ตามลำดับ และในชุดที่ใช้ทดสอบจริงอยู่ที่ 88.82, 98.48, และ 93.40 ตามลำดับ

ปัญหาที่เกิดจากการสกัด NE ที่พบในการพัฒนาระบบนั้น ก็คือปัญหาที่เกิดจากการสกัดคำที่มีความหมายกำกวม เช่น “สามารถ”, “แพร่”, “รายงาน” เป็นต้น ซึ่งคำเหล่านี้สามารถมีความหมายได้หลายความหมาย

และสำหรับประสิทธิภาพในการสกัดความสัมพันธ์นั้น ในชุดข้อความฝึกฝนระบบนั้นมีค่า Recall, Precision, และค่า F อยู่ที่ 88.21, 94.51, และ 91.25 ตามลำดับ และในชุดที่ใช้ทดสอบจริงอยู่ที่ 81.33, 89.05, และ 85.02 ตามลำดับ

ถึงแม้จะมีการสร้างกฎเพื่อให้ระบบเข้าใจลักษณะความสัมพันธ์ที่อยู่ในข้อความที่มีการเขียนในรูปแบบต่างๆ แต่ในการสกัดความสัมพันธ์ในบางข้อความก็ยังมีความผิดพลาดเกิดขึ้นได้ เนื่องมาจากความผิดพลาดที่เกิดขึ้นนั้น เกิดขึ้นมาจากการที่ระบบไม่มีความเข้าใจในลักษณะและความหมายของ NE ต่างๆ รวมไปถึงไม่มีความเข้าใจในบริบทต่างๆ ภายในข้อความ จึงทำให้ระบบไม่เข้าใจ หรืออาจจะเข้าใจผิดในความหมายของข้อความนั้นๆ ได้

อย่างไรก็ตามถึงแม้ว่าระบบสกัดความสัมพันธ์นี้จะไม่มีความสามารถที่จะเข้าใจความหมายของข้อความได้อย่างครบถ้วน แต่วิธีการสกัดความสัมพันธ์โดยการสร้างกฎและใช้คำสำคัญนี้ก็ยังมีข้อดีในแง่ของการพัฒนาระบบ คือใช้เพียงคำสำคัญจำนวนไม่มากในการพัฒนาระบบเพื่อให้สามารถสกัดความสัมพันธ์ได้ และยังสามารถส่งผลถึงระยะเวลาที่ใช้ในการประมวลผลที่เร็วกว่าการตรวจสอบจากคลังข้อมูลจำนวนมากอีกด้วย ซึ่งจากจำนวนคำสำคัญที่นำมาใช้ในการหาคำสำคัญ คือ ใช้กริยาจำนวนเพียง 48 คำ (แบ่งเป็นชนิด create 22 คำ, goto 10 คำ, และ located_in 16 คำ) และใช้คำบุพบทจำนวนเพียง 22 คำ (แบ่งเป็นชนิด create 5 คำ, goto 6 คำ, located_in 10 คำ, และคำที่บุพบทที่สามารถใช้กับทุกชนิดอีก 1 คำ) ซึ่งเมื่อแบ่งย่อยเป็นชนิดต่างๆ แล้ว จะพบว่าเป็นจำนวนที่ค่อนข้างน้อย แต่สามารถที่จะนำมาสกัดความสัมพันธ์จากข้อความจำนวนมากได้ ซึ่งเป็นเพราะว่าในการกล่าวถึงความสัมพันธ์ชนิดใดก็ตามในภาษามนุษย์นั้น มีคำที่สามารถบ่งบอกถึง

ความสัมพันธ์แต่ละชนิดอยู่จำนวนไม่กี่คำเท่านั้น แต่สิ่งที่ทำให้เกิดความหลากหลายในภาษามนุษย์นั้นคือการนำคำต่างๆ มาผสมกันเพื่อให้เกิดความหมายที่แตกต่างกันไปนั่นเอง

ข้อเสนอแนะ

จากการที่ระบบสกัดความสัมพันธ์ที่พัฒนาขึ้นมานี้ยังมีปัญหาในเรื่องความเข้าใจความหมายของข้อความดังที่กล่าวมา ซึ่งถือเป็นข้อด้อยของระบบนี้ แต่เนื่องจากข้อดีในแง่ของการพัฒนาระบบซึ่งใช้เพียงคำสำคัญเพียงไม่กี่คำในหนึ่งความสัมพันธ์ รวมไปถึงผลลัพธ์ที่ได้ก็อยู่ในระดับที่ยอมรับได้ ซึ่งหากในอนาคตสามารถสร้างระบบนี้ให้สามารถที่จะเพิ่มคำสำคัญใหม่ๆ เข้าไปได้ ระบบนั้นก็จะเป็นระบบที่สามารถที่จะรู้จักความสัมพันธ์ใหม่ๆ ได้ตลอดเวลา เนื่องจากในแต่ละความสัมพันธ์นั้นจะมีคำที่สามารถบ่งบอกถึงความสัมพันธ์อยู่จำนวนไม่มาก ไม่จำเป็นต้องใช้คลังข้อมูลจำนวนมากในการเรียนรู้ จึงมีความเป็นไปได้ที่ผู้ใช้จะสามารถเพิ่มความสัมพันธ์ใหม่ๆ ให้ระบบรู้จักได้โดยง่าย ดังนั้นวิธีการใช้กฎและคำสำคัญในการค้นหาความสัมพันธ์จึงยังมีความน่าสนใจที่จะนำมาใช้ แต่เพื่อที่จะเพิ่มประสิทธิภาพของระบบให้มากขึ้นตลอดจนให้มีความครบถ้วนในการที่จะสามารถเข้าลักษณะการเกิดความสัมพันธ์ในข้อความรูปแบบต่างๆ จึงขอเสนอว่าในอนาคตควรจะนำวิธีการประมวลผลทางภาษาธรรมชาติชนิดอื่นๆ เข้ามาร่วมด้วยในอนาคต เช่น

- การวิเคราะห์เชิงโครงสร้าง (Syntactic Analysis) เพื่อให้สามารถที่จะวิเคราะห์โครงสร้างของประโยคได้ว่า คำต่างๆ ในประโยคนั้นมีหน้าที่ใดบ้าง ซึ่งจะมีส่วนช่วยในการค้นหาคำสำคัญต่างๆ เนื่องจากจะทำให้ระบบสามารถเข้าใจได้ว่าคำสำคัญที่ค้นพบในประโยคนั้น มีหน้าที่ตรงกับคำที่ต้องการค้นหาหรือไม่ เช่น คำกริยา หรือคำบุพบท เป็นต้น

- การวิเคราะห์เชิงความหมาย (Semantic Analysis) เพื่อให้ระบบสามารถที่ทราบความหมายของข้อความที่ต้องการวิเคราะห์ได้ ดังจะเห็นได้จากปัญหาของระบบนี้ที่ในบางครั้งไม่สามารถที่จะเข้าใจความหมายในบริบทของข้อความว่ากำลังต้องการสื่ออะไร ทำให้ในบางครั้งเกิดความเข้าใจผิดได้หากพบคำสำคัญในประโยคที่ไม่ได้มีความสัมพันธ์

- การวิเคราะห์เชิงตีความ (Pragmatic Analysis) เพื่อให้สามารถเข้าใจความหมายที่แฝงอยู่ในข้อความได้ ยกตัวอย่างเช่น “เมื่อคุณติดไวรัสโทรจัน” จะเป็นได้ว่าคำว่า “คุณ” ในที่นี้ไม่ได้หมายความว่า “คุณ” ที่เป็นมนุษย์ได้ “ติดไวรัสโทรจัน” หากแต่เป็น “เครื่อง(ของคุณ)”

ต่างหากที่ “ติดไวรัสโทรจัน” จากลักษณะเช่นนี้จะมีประโยชน์มากหากระบบสามารถที่จะตีความความหมายที่แฝงอยู่ในข้อความได้

นอกจากนี้ยังแนะนำให้ควรมีการปรับปรุงกฎบางอย่างที่ใช้ในการสกัดความสัมพันธ์ เช่น ระยะห่างของคำสำคัญกับ NE ทางด้านซ้ายและขวา, ความสัมพันธ์ที่มีการกำหนดประเภท NE ทางด้านซ้าย และ ความสัมพันธ์ที่เกิดขึ้นในบริบทหลังจากบริบทปัจจุบัน เนื่องมาจากผู้วิจัยได้ทำการทดสอบกฎเหล่านี้กับชุดข้อความที่ได้นำมาฝึกฝนกับระบบซึ่งมีจำนวนน้อย จึงอาจทำให้ผลลัพธ์ที่ได้ยังไม่สมบูรณ์ที่สุดหากมีการนำไปใช้กับข้อความอื่นๆ จึงขอแนะนำให้มีการปรับปรุงกฎเหล่านี้ว่าควรจะมีระยะในการตรวจสอบเท่าใด จึงจะให้ผลลัพธ์ที่ถูกต้องที่สุด

บรรณานุกรม

ภาษาไทย

กระทรวงพลังงาน. สำนักงานนโยบายและแผนพลังงาน. รวมเว็บราชการไทย [ออนไลน์]. เข้าถึงเมื่อ 2 กันยายน 2551. เข้าถึงได้จาก http://www.eppo.go.th/index_thaigov-T.html#1

จำเรียง จันทรประภา, ผู้รวบรวม. ชื่อทะเล [ออนไลน์]. เข้าถึงเมื่อ 6 สิงหาคม 2551. เข้าถึงได้จาก http://www.royin.go.th/th/profile/index.php?PageNo=1&PageShow=221&SystemModuleKey=245

ตลาดหลักทรัพย์แห่งประเทศไทย. รายชื่อบริษัทจดทะเบียนในตลาดหลักทรัพย์ [ออนไลน์]. เข้าถึงเมื่อ 2 กันยายน 2551. Available from http://www.set.or.th/th/company/files/listed_company8_4_52_TE.xls

ธนาคารแห่งประเทศไทย. รายชื่อสถาบันการเงิน [ออนไลน์]. เข้าถึงเมื่อ 9 กันยายน 2551. เข้าถึงได้จาก http://www.bot.or.th/Thai/FinancialInstitutions/WebsiteFI/Pages/Name.aspx

บ้านจอมยุทธ. ประเทศไทย 76 จังหวัด (ประวัติศาสตร์ สาระความรู้ ข้อมูล แหล่งท่องเที่ยว) [ออนไลน์]. เข้าถึงเมื่อ 29 สิงหาคม 2551. เข้าถึงได้จาก http://www.baanjomyut.com/index.html

ร่างประกาศสำนักนายกรัฐมนตรี และประกาศราชบัณฑิตยสถาน. “กำหนดชื่อประเทศ ดินแดน เขตการปกครอง และเมืองหลวง.” 22 กุมภาพันธ์ 2544.

ราชบัณฑิตยสถาน. ลำดับชื่อจังหวัด เขต อำเภอ และกิ่งอำเภอ [ออนไลน์]. เข้าถึงเมื่อ 6 สิงหาคม 2551. เข้าถึงได้จาก http://www.royin.go.th/upload/246/FileUpload/417_4191.pdf

สุฤดี ฉัตรไตรมงคล. “การรู้จำและการจำแนกประเภทของชื่อเฉพาะภาษาไทย.” วิทยานิพนธ์ปริญญาอักษรศาสตร์มหาบัณฑิต สาขาวิชาภาษาศาสตร์ คณะอักษรศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย, 2548.

Arip. ข่าว IT [ออนไลน์]. เข้าถึงเมื่อ 10 สิงหาคม 2551. เข้าถึงได้จาก http://www.arip.co.th/2006/news_list.php

Nook [นามแฝง]. Orchid Corpus [ออนไลน์]. 5 มิถุนายน 2551. Available from http://nookplanet.blogspot.com/2008/06/orchid-corpus.html

Yopi. รายชื่อยี่ห้อทั้งหมด [ออนไลน์]. เข้าถึงเมื่อ 3 สิงหาคม 2551. เข้าถึงได้จาก http://www.yopi.co.th/mfr_idx/

ภาษาอังกฤษ

Agichtein, Eugene, and Luis Gravano. “Snowball: Extracting Relations from Large Plain-Text Collections.” In Proceedings of the Fifth ACM Conference on Digital Libraries, 85-94. New York : ACM, 2000.

Appelt , Douglas E. et al. “FASTUS: a Finite-state Processor for Information Extraction from Real-world Text.” In Proceedings of the 13th International Joint Conference on Artificial Intelligence, 1179-1185. Chambéry : Morgan Kaufmann, 1993.

Baluja, Shumeet, Vibhu O. Mittal, and Rahul Sukthankar. “Applying Machine Learning for High Performance Named-Entity Extraction.” Computational Intelligence 16, 4 (November 2000) : 586-596.

Bikel, Daniel M. et al. “Nymble: a High-Performance Learning Name-finder.” In Proceedings of the Fifth Conference on Applied Natural Language Processing, 194-201. Morristown : Association for Computational Linguistics, 1997.

Borthwick, Andrew et al. “NYU: Description of the MENE Named Entity System as Used in MUC-7.” In Proceedings of the 7th Message Understanding Conference (MUC-7). n.p., 1998.

Brin, Sergey. “Extracting Patterns and Relations from World Wide Web.” In Conjunction with EDBT'98, 172-183. Valencia : Springer, 1998.

Chanlekha, Hutchatai, and Asanee Kawtrakul. “Thai Named Entity Extraction by incorporating Maximum Entropy Model with Simple Heuristic Information**.”** In IJCNLP’ 2004. n.p., 2004.

Charoenpornsawat, Paisarn, Boonserm Kijsirikul, and Surapant Meknavin. “Feature-based Proper Name Identification in Thai.” In Proc. of National Computer Science and Engineering Conference’98 (NCSEC’98). n.p., 1998.

Chinchor, Nancy A. OVERVIEW OF MUC-7/MET-2 [Online]. Accessed 5 February 2008. Available from http://www.itl.nist.gov/ iaui/894.02/related_projects/muc/proceedings/ muc_7_proceedings/overview.html

Collins, Michael, and Yoram Singer, "Unsupervised Models For Named Entity Classification." In Proc. of the 1999 Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora, 100-110. Morristown : ACL, 1999.

Cucerzan, Silviu, and David Yarowsky, "Language Independent Named Entity Recognition Combining Morphological and Contextual Evidence." In Proc. of the 1999 Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora, 90-99. Morristown : ACL, 1999.

Culotta, Aron and Jeffrey Sorensen. "Dependency Tree Kernels for Relation Extraction." In Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics, 423-429. Morristown : ACL, 2004.

Hasegawa, Takaaki, Satoshi Sekine, and Ralph Grishman. "Discovering Relations among Named Entities from Large Corpora." In Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics, 415-422. Morristown : ACL, 2004.

Gaizauskas, R. et al. "University of Sheffield: Description of the LaSIE system as used for MUC-6." In Proc. of the sixth Message Understanding Conference (MUC-6), 207-220. Morristown : ACL, 1995.

Giuliano, Claudio, Alberto Lavelli, and Lorenza Romano, "Relation Extraction and the Influence of Automatic Named-Entity Recognition." ACM Transactions on Speech and Language Processing (TSLP) 5, 1 (December 2007) : 2:1-2:26.

Kawtrakul, Asanee et al. "Automatic Thai Unknown Word Recognition." In NLPRS 97, 341-348. n.p., 1997.

Kijsirikul, Boonserm, Paisarn Charoenpornsawat, and Surapant Meknavin. “Comparing Winnow and RIPPER in Thai Named-Entity Identification.” In Proc. of the Natural Language Processing Pacific Rim Symposium 1999 (NLPRS’99). n.p., 1999.

Lin, Dekang, and Patrick Pantel. “Dirt - Discovery of Inference Rules from Text.” In Proceedings of ACM Conference on Knowledge Discovery and Data Mining (KDD-01), 323-328. New York : ACM, 2001.

Microsoft. List of antivirus software vendors [Online]. Updated 14 April 2008. Available from http://support.microsoft.com/kb/49500

Mikheev, Andrei, Claire Grover, and Marc Moens. “Description of the LTG System Used for MUC-7.” In Proc. of 7th Message Understanding Conference (MUC-7). n.p., 1998.

MUC-6 [Online]. Updated 25 April 1996. Available from http://cs.nyu.edu/cs/faculty/grishman/muc6.html

Named Entity Task Definition [Online]. Updated 2 June 1995. Available from http://www.cs.nyu.edu/ cs/faculty/ grishman/NEtask20.book_1.html

Orchid Corpus [Online]. Accessed 14 February 2008. Available from http://www.links.nectec.or.th/orchid/

Ravichandran, Deepak, and Eduard Hovy. “Learning Surface Text Patterns for a Question Answering System.” in Proceedings of the ACL Conference, 41-47. Morristown : ACL, 2002.

Roth, Dan, and Wen-tau Yih. “Probabilistic Reasoning for Entity & Relation Recognition.” In Proceedings of the 19th International Conference on Computational Linguistics (COLING’02), 1-7. Morristown : ACL, 2002.

_____ . “A Linear Programming Formulation for Global Inference in Natural Language Tasks.” In Proceedings of the 8th Conference on Computational Natural Language Learning (CoNLL’04), 1-8. n.p., 2004.

Roth, Dan, and Wen-tau Yih. “Global Inference for Entity and Relation Identification via a Linear Programming Formulation.” In Introduction to Statistical Relational Learning, 553-576. Edited by Lise Getoor and Ben Taskar. Cambridge : The MIT Press, 2007.

Sekine, Satoshi, Ralph Grishman, and Hiroyuki Shinnou. “A Decision Tree Method for Finding and Classifying Names in Japanese Texts.” In Proceedings of the Sixth Workshop on Very Large Corpora, 171-178. New Brunswick : ACL, 1998.

Sekine, Satoshi. Definition of Sekine’s Extended Named Entity [Online]. Updated 29 October 2003. Available from http://nlp.cs.nyu.edu/ene/version6_1_0eng.html

_____ . Sekine’s Extended Named Entity Hierarchy [Online]. Updated 23 March 2007. Available from http://nlp.cs.nyu.edu/ene/

Software: KU Wordcut [Online]. Accessed 23 September 2008. Available from http://naist.cpe.ku.ac.th/pkg/kucut-1.2.2_python25_fix.zip

Stevenson, Mark, and Robert Gaizauskas. “Using Corpus-derived Name Lists for Named Entity Recognition.” In Proceedings of the Sixth Applied Natural Language Processing Conference and the First Meeting of the North American Chapter of the Association for Computational Linguistics, 290-295. Morristown : ACL, 2000.

The ACE 2005 (ACE05) Evaluation Plan [Online]. Updated 3 October 2005. Available from http://www.nist.gov/speech/tests/ace/ace05/doc/ace05-evalplan.v3.pdf

Wacholder, Nina, Yael Ravin, and Misook Choi. “Disambiguation of Proper Names in Text.” In Proceedings of the 5th Applied Natural Language Processing Conference, 202-208. Morristown : ACL, 1997.

Winaddons.com. Top 300 Freeware Software! [Online]. Updated 7 January 2007. Available from http://www.winaddons.com/top-300-freeware-software/

Zelenko, Dmitry, Chinatsu Aone, and Anthony Richardella. “Kernel Methods for Relation Extraction,” Journal of Machine Learning Research 3, (March 2003) : 1083-1106.

ภาคผนวก

ภาคผนวก ก

ข้อมูลที่ใช้ในการพัฒนาระบบ

ตารางที่ 10 แบบแผนของคำสำคัญชนิดคำกริยา

number	word	type	ne1	unwant_ne1	ne2	unwant_ne2	unwant_pair_ne1	unwant_pair_ne2	direction	unwant	want	want_after_ne2	adjacent	ncr_want	may_want	len_to_ne2	len_from_ne1
1	ผลิต	create				per,org							0	0		1	0
2	จัดตั้ง	create											0	0		1	0
3	ก่อให้เกิด	create											0	0		1	0
4	ทำให้เกิด	create											0	0		1	0
5	ส่ง	create	org		pro								0	0		1	0
6	ออกแบบ	create											1	0		1	0
7	จัดทำ	create											1	0		1	0
8	ออก	create	org		pro								1	0		1	0
9	พิมพ์	create											1	0		1	0
10	คิดค้น	create											0	0		1	0
11	บัญญัติ	create											0	0		1	0
12	ก่อตั้ง	create											0	0		1	0
13	ออก	create	org		pro								1	0		2	0
14	สร้าง	create											1	0	prep	1	0
15	เกิด	create							left		prep		1	0		2	0
16	เปิดตัว	create	org		pro								1	0		1	0
17	พัฒนา	create							left		ด้วย ,โดย		1			3	0
18	ทำขึ้น	create									.prep		0	0		1	0

ตารางที่ 10 (ต่อ)

number	word	type	ne1	unwant_ne1	ne2	unwant_ne2	unwant_pair_ne1	unwant_pair_ne2	direction	unwant	want	want_after_ne2	adjacent	ncr_want	may_want	len_to_ne2	len_from_ne1
19	เขียน	create	per		pro								1	0		1	0
20	ทำ	create		loc			pro	per					1			1	0
21	พัฒนา	create	per,org										1	0	prep	1	0
22	เขียน	create	pro						left		ด้วย,โดย		1			3	0
23	ส่ง	goto		org,loc						.ด้วย,.โดย		ไป,ให้	1	1	ให้,ไป (2)	1	0
24	เข้า	goto								นำ.			1	0	prep	2	0
25	มา	goto		loc						จาก,ห.,สา.,.ก,ผ่าน.,.,.ก			1	0	.prep	1	0
26	ไป	goto		loc						.จนถึง,.จาก,.ด้วย,.พบ,.แล้ว,ต่อ.,ทั่ว.			1	0	prep	2	0

ตารางที่ 10 (ต่อ)

number	word	type	ne1	nt_ne	ne2	nt_ne	unwant_pair_ne1	unwant_pair_ne2	direction	unwant	want	want_after_ne2	adjacent	ncr_want	may_want	len_to_ne2	len_from_ne1
27	ไปพบ	goto	per	loc	per								1	0		1	0
28	รับ	goto							left	สำห.,ป.			1	0		1	0
29	เยี่ยม	goto	per										1	0	.ชม	1	0
30	ลง	goto	pro		pro					แป.,.มือ			1	0	prep	1	0
31	คืน	goto										ให้	1	1	ให้	1	0
32	แจก	goto										ให้	1	1		2	0
33	เก็บ	located_in							left				1	0	prep	1	0
34	วาง	located_in									prep		0	0		1	0
35	ติดตั้ง	located_in		per					left		ที่,ซึ่ง,ได้		1	0		1	0
36	ใส่	located_in							left				1	0		1	1
37	อยู่	located_in								ขึ้น.	prep		1	0		2	0
38	ตั้ง	located_in									prep		0	0		1	0
39	ฝัง	located_in									prep		0	0		1	0

ตารางที่ 10 (ต่อ)

number	word	type	ne1	unwant_ne1	ne2	unwant_ne2	unwant_pair_ne1	unwant_pair_ne2	direction	unwant	want	want_after_ne2	adjacent	ncr_want	may_want	len_to_ne2	len_from_ne1
40	มี	located_in		per					left	.ต่อ,.ใน			1	0		2	0
41	อาศัย	located_in									prep		0	0		1	0
42	ติดตั้ง	located_in		per								ไว้	1	0	.ให้	1	0
43	ห่อหุ้ม	located_in							left				0	0		1	0
44	ใส่	located_in										ไว้	0	1		1	0
45	ติด	located_in			pro				left	.ตั้ง			1	0		1	0
46	รวม	located_in										ไว้	1	0	ไว้	1	0
47	ประกอบด้วย	located_in							left				1			1	0
48	บรรจุ	located_in							left				1	0		1	0

ตารางที่ 11 แบบแผนของคำสำคัญชนิดคำบุพบท

number	word	type	ne1	unwant_ne1	ne2	unwant_ne2	unwant_pair_ne1	unwant_pair_ne2	direction	unwant	want	want_after_ne2	adjacent	ncr_want	may_want	len_to_ne2	len_from_ne1
1	ที่										verb		1	0		1	0
2	จาก	create	pro		org				left				1	0		1	0
3	โดย	create			org,per,pro				left		verb		1	0		2	0
4	จาก	create							left		verb		1	0		2	0
5	โดย	create			org,per				left				1	0		2	0
6	ด้วย	create		per					left		verb		1	0		1	0
7	สู่	goto											0	0		1	0
8	ถึง	goto									verb		1	0		1	0
9	ให้	goto								ทำ.	verb		1	0		2	0
10	ใน	goto								รูปของ,รูปแบบ	verb		1	0		2	0
11	ยัง	goto									verb		1	0		1	0
12	บน	goto									verb		1	0		1	0

ตารางที่ 11 (ต่อ)

number	word	type	ne1	unwant_ne1	ne2	unwant_ne2	unwant_pair_ne1	unwant_pair_ne2	direction	unwant	want	want_after_ne2	adjacent	ncr_want	may_want	len_to_ne2	len_from_ne1
13	ใน	located_in					org	pro		นอก, รูป ของ, ที่สุด ,. ขณะ			1	0		2	0
14	บน	located_in					org,loc	pro					1	0		1	0
15	หน้า	located_in					per	per			verb		0	0		1	0
16	ข้าง	located_in					per	per			verb		0	0		1	0
17	หลัง	located_in					per	per			verb		0	0		1	0
18	ใต้	located_in				per					verb		0	0		1	0
19	กลาง	located_in					per	per			verb		0	0		1	0
20	จาก	located_in	pro		pro					เนื่อง, นอก. ,.นี้,. นั้น			1	0		1	1
21	เติมไปด้วย	located_in							left				1	0		1	0
22	ที่	located_in											1	0	.ด้าน	1	1

ตารางที่ 12 คำบุพบทที่แสดงถึงความเป็นเจ้าของ

word	unwant
ของ	
จาก	นอก.,.นี้,.นั้น
เพื่อ	
บน	
ภายใต้	
สำหรับ	
หน้า	
นอก	.จาก,.เหนือ
หลัง	
ข้างหลัง	
รอบๆ	
ข้างบน	
ในรูปของ	
ใน	
ประเภท	
ที่	.ว่า
เท่า	.นั้น
ในทาง	

ตารางที่ 13 คำที่แสดงถึงประโยคปฏิเสธ

word	unwant
ไม่	.เพียง,.ว่า,.เกิน,.มาก ,.น้อย,.หวัง,ความ.
มิใช่	
มิได้	

ภาคผนวก ข

ผลลัพธ์ของระบบ

ตารางที่ 14 ผลการสกัด NE ในชุดฝึกฝน (Training Set)

ข้อความ	สกัด NE ได้		สกัด NE ไม่ได้		รวม NE ที่สกัดไม่ได้	รวม NE ทั้งหมดในข้อความ
	NEที่สกัดได้ถูกต้อง	สกัดได้แต่ผิด	ผิดพลาดจากการตัดคำ	ระบบไม่รู้จักหรือผิดพลาด		
1	10					10
2	9					9
3	10			1	1	11
4	9			1	1	10
5	10					10
6	10					10
7	10					10
8	16		4	1	5	21
9	13		2		2	15
10	10					10
11	7		1		1	8
12	6			1	1	7
13	6			1	1	7
14	10		1		1	11
15	7			2	2	9
16	10					10
17	11					11
18	7					7
19	11		1		1	12
20	12					12
21	5					5
22	14		1		1	15

ตารางที่ 14 (ต่อ)

ข้อความ	สกัด NE ได้		สกัด NE ไม่ได้		รวม NE ที่สกัดไม่ได้	รวม NE ทั้งหมดในข้อความ
	NEที่สกัดได้ถูกต้อง	สกัดได้แต่ผิด	ผิดพลาดจากการตัดคำ	ระบบไม่รู้จักหรือผิดพลาด		
23	5		3		3	8
24	4					4
25	15		1	1	2	17
26	9			1	1	10
27	8					8
28	10		1		1	11
29	5					5
30	8			3	3	11
31	4					4
32	5			1	1	6
33	9			1	1	10
34	6					6
35	8		1		1	9
36	11			1	1	12
37	8					8
38	9					9
39	8					8
40	4					4
41	9	1		3	3	12
42	4	1	2	1	3	7
43	8					8
44	12		1	1	2	14

ตารางที่ 14 (ต่อ)

ข้อความ	สกัด NE ได้		สกัด NE ไม่ได้		รวม NE ที่สกัดไม่ได้	รวม NE ทั้งหมดในข้อความ
	NEที่สกัดได้ถูกต้อง	สกัดได้แต่ผิด	ผิดพลาดจากการตัดคำ	ระบบไม่รู้จักหรือผิดพลาด		
45	11			1	1	12
46	6					6
47	13					13
48	10		1		1	11
49	7					7
50	9			1	1	10
51	9			1	1	10
52	9					9
53	11			4	4	15
54	6			2	2	8
55	9					9
56	5					5
57	10					10
58	12		2	1	3	15
59	6					6
60	10		1		1	11
61	7					7
62	5	1				5
63	9					9
64	6					6
65	8					8
66	4					4

ตารางที่ 14 (ต่อ)

ข้อความ	สกัด NE ได้		สกัด NE ไม่ได้		รวม NE ที่สกัดไม่ได้	รวม NE ทั้งหมดในข้อความ
	NEที่สกัดได้ถูกต้อง	สกัดได้แต่ผิด	ผิดพลาดจากการตัดคำ	ระบบไม่รู้จักหรือผิดพลาด		
67	10		1		1	11
68	8					8
69	5		1	1	2	7
70	8					8
71	10			1	1	11
72	7					7
73	6					6
74	12					12
75	10			1	1	11
76	9		1		1	10
77	10					10
78	14	3				14
79	4	1				4
80	8					8
81	14					14
82	6		1		1	7
83	7		1	2	3	10
84	11		1	1	2	13
85	12			1	1	13
86	7					7
87	6					6
88	6		1		1	7

ตารางที่ 14 (ต่อ)

ข้อความ	สกัด NE ได้		สกัด NE ไม่ได้		รวม NE ที่สกัดไม่ได้	รวม NE ทั้งหมดในข้อความ
	NEที่สกัดได้ถูกต้อง	สกัดได้แต่ผิด	ผิดพลาดจากการตัดคำ	ระบบไม่รู้จักหรือผิดพลาด		
89	7					7
90	9	2		2	2	11
91	8			2	2	10
92	7	1				7
93	6					6
94	4		1	1	2	6
95	7			1	1	8
96	5			1	1	6
97	8					8
98	7					7
99	7	2				7
100	4	1				4
รวม	**833**	**13**	**31**	**44**	**75**	**908**
NE ที่ระบบเลือกมาทั้งหมด						**846**
NE ที่ถูกต้องทั้งหมดในเอกสาร(ไม่รวมปัญหาตัดคำ)						**877**

ตารางที่ 15 ผลการสกัด NE ในชุดทดสอบจริง (Test Set)

ข้อความ	สกัด NE ได้		สกัด NE ไม่ได้		รวม NE ที่สกัดไม่ได้	รวม NE ทั้งหมดในข้อความ
	Neที่สกัดได้ถูกต้อง	สกัดได้แต่ผิด	ผิดพลาดจากการตัดคำ	ระบบไม่รู้จักหรือผิดพลาด		
1	9		1		1	10
2	10			1	1	11
3	13					13
4	5					5
5	11	1		3	3	14
6	10	3		1	1	11
7	12			2	2	14
8	8	1		2	2	10
9	5			1	1	6
10	3					3
11	7			3	3	10
12	9	1		1	1	10
13	10			1	1	11
14	9		2	1	3	12
15	7			1	1	8
16	5			3	3	8
17	6			3	3	9
18	6			2	2	8
19	10		1		1	11
20	9		2		2	11
21	9			2	2	11
22	9		2		2	11
23	10		2	1	3	13

ตารางที่ 15 (ต่อ)

ข้อความ	สกัด NE ได้		สกัด NE ไม่ได้		รวม NE ที่สกัดไม่ได้	รวม NE ทั้งหมดในข้อความ
	Neที่สกัดได้ถูกต้อง	สกัดได้แต่ผิด	ผิดพลาดจากการตัดคำ	ระบบไม่รู้จักหรือผิดพลาด		
24	8		1	1	2	10
25	8	1	1		1	9
26	13		1	2	3	16
27	6		1		1	7
28	7					7
29	5			3	3	8
30	8			1	1	9
31	8		1	1	2	10
32	8					8
33	13		3	2	5	18
34	9		1	1	2	11
35	10		1	1	2	12
36	6					6
37	7		1	2	3	10
38	6			1	1	7
39	7					7
40	8			1	1	9
41	3					3
42	7					7
43	6			1	1	7
44	10					10
45	5					5
46	8		3		3	11

ตารางที่ 15 (ต่อ)

ข้อความ	สกัด NE ได้		สกัด NE ไม่ได้		รวม NE ที่สกัดไม่ได้	รวม NE ทั้งหมดในข้อความ
	Neที่สกัดได้ถูกต้อง	สกัดได้แต่ผิด	ผิดพลาดจากการตัดคำ	ระบบไม่รู้จักหรือผิดพลาด		
47	9					9
48	8		1		1	9
49	9		1		1	10
50	8		1		1	9
51	9			1	1	10
52	10		1		1	11
53	4			1	1	5
54	8					8
55	5		1	1	2	7
56	12	1		3	3	15
57	8			3	3	11
58	8			2	2	10
59	11		2		2	13
60	8			7	7	15
61	5		1	1	2	7
62	5		1	1	2	7
63	4			1	1	5
64	9		1	1	2	11
65	6					6
66	4			3	3	7
67	11					11
68	7		1		1	8
69	6			1	1	7

ตารางที่ 15 (ต่อ)

ข้อความ	สกัด NE ได้		สกัด NE ไม่ได้		รวม NE ที่สกัดไม่ได้	รวม NE ทั้งหมดในข้อความ
	Neที่สกัดได้ถูกต้อง	สกัดได้แต่ผิด	ผิดพลาดจากการตัดคำ	ระบบไม่รู้จักหรือผิดพลาด		
70	6	1	5	1	6	12
71	10	1	2	2	4	14
72	4					4
73	9		3	1	4	13
74	13		1	1	2	15
75	10		1		1	11
76	10			2	2	12
77	8		2	1	3	11
78	11					11
79	8		1		1	9
80	10		2		2	12
81	7		1	1	2	9
82	8					8
83	3			1	1	4
84	7		1	2	3	10
85	5			1	1	6
86	7	2				7
87	8	1				8
88	10		1		1	11
89	7			1	1	8
90	3			2	2	5
91	4			2	2	6
92	6					6

ตารางที่ 15 (ต่อ)

ข้อความ	สกัด NE ได้		สกัด NE ไม่ได้		รวม NE ที่สกัดไม่ได้	รวม NE ทั้งหมดในข้อความ
	Neที่สกัดได้ถูกต้อง	สกัดได้แต่ผิด	ผิดพลาดจากการตัดคำ	ระบบไม่รู้จักหรือผิดพลาด		
93	5					5
94	5					5
95	7					7
96	9					9
97	8			1	1	9
98	5		1		1	6
99	6		3		3	9
100	8		1		1	9
101	6			1	1	7
102	8		1		1	9
103	8		1		1	9
104	3			1	1	4
105	7	1				7
106	4		2	1	3	7
107	7			1	1	8
108	5		2	1	3	8
109	5			3	3	8
110	4			4	4	8
111	7					7
112	8	1		1	1	9
113	10					10
114	3			2	2	5
115	8					8

ตารางที่ 15 (ต่อ)

ข้อความ	สกัด NE ได้		สกัด NE ไม่ได้		รวม NE ที่สกัดไม่ได้	รวม NE ทั้งหมดในข้อความ
	Neที่สกัดได้ถูกต้อง	สกัดได้แต่ผิด	ผิดพลาดจากการตัดคำ	ระบบไม่รู้จักหรือผิดพลาด		
116	6			2	2	8
117	7					7
118	6		3	1	4	10
119	6		2		2	8
120	5			1	1	6
121	5		2	3	5	10
122	5					5
123	4			2	2	6
124	4			1	1	5
125	6	1	2		2	8
126	4		3	1	4	8
127	2		3		3	5
128	5					5
129	8					8
130	5		1		1	6
131	6			1	1	7
132	4		1	2	3	7
133	11			1	1	12
134	6			1	1	7
135	8					8
136	8			1	1	9
137	3					3
138	5					5

ตารางที่ 15 (ต่อ)

ข้อความ	สกัด NE ได้		สกัด NE ไม่ได้		รวม NE ที่สกัดไม่ได้	รวม NE ทั้งหมดในข้อความ
	Neที่สกัดได้ถูกต้อง	สกัดได้แต่ผิด	ผิดพลาดจากการตัดคำ	ระบบไม่รู้จักหรือผิดพลาด		
139	8					8
140	9					9
141	6		1		1	7
142	6					6
143	6					6
144	4		1		1	5
145	7			1	1	8
146	11		1		1	12
147	5			1	1	6
148	8					8
149	6					6
150	5		1	1	2	7
151	5		1	1	2	7
152	4		2	1	3	7
153	4			3	3	7
154	8	1	1	1	2	10
155	11					11
156	8					8
157	8					8
158	2		1	1	2	4
159	7	1		2	2	9
160	6		1	1	2	8
161	6			2	2	8

ตารางที่ 15 (ต่อ)

ข้อความ	สกัด NE ได้		สกัด NE ไม่ได้		รวม NE ที่สกัดไม่ได้	รวม NE ทั้งหมดในข้อความ
	Neที่สกัดได้ถูกต้อง	สกัดได้แต่ผิด	ผิดพลาดจากการตัดคำ	ระบบไม่รู้จักหรือผิดพลาด		
162	8					8
163	4			1	1	5
164	7			2	2	9
165	10			3	3	13
166	5					5
167	5					5
168	4					4
169	4					4
170	8			2	2	10
171	3			3	3	6
172	4					4
173	5			1	1	6
174	6			1	1	7
175	4			1	1	5
176	5					5
177	12	1		1	1	13
178	4			2	2	6
179	3		1	1	2	5
180	6		2	3	5	11
181	6	1	1	2	3	9
182	4			2	2	6
183	13			2	2	15
184	4					4

ตารางที่ 15 (ต่อ)

ข้อความ	สกัด NE ได้		สกัด NE ไม่ได้		รวม NE ที่สกัดไม่ได้	รวม NE ทั้งหมดในข้อความ
	Neที่สกัดได้ถูกต้อง	สกัดได้แต่ผิด	ผิดพลาดจากการตัดคำ	ระบบไม่รู้จักหรือผิดพลาด		
185	7					7
186	5					5
187	5			1	1	6
188	5					5
189	3			1	1	4
190	2		2		2	4
191	4		1	1	2	6
192	7			1	1	8
193	7			1	1	8
194	7		1		1	8
195	7			2	2	9
196	10		1		1	11
197	4	1	1		1	5
198	6					6
199	1		1		1	2
200	4		1		1	5
รวม	**1359**	**21**	**105**	**171**	**276**	**1635**
NE ที่ระบบเลือกมาทั้งหมด						**1380**
NE ที่ถูกต้องทั้งหมดในเอกสาร(ไม่รวมปัญหาตัดคำ)						**1530**

ตารางที่ 16 ผลการสกัดความสัมพันธ์ในชุดฝึกฝน (Training Set)

ข้อความ	สกัดได้		สกัดความสัมพันธ์ไม่ได้			รวมความสัมพันธ์ที่สกัดไม่ได้	ความสัมพันธ์ทั้งหมดในข้อความ
	ความสัมพันธ์ที่สกัดได้ถูก	สกัดได้แต่ผิด	ปัญหาจากการสกัด NE	ปัญหาจากการตัดคำ	ไม่เข้าใจความหมายหรือผิดพลาด		
1	1						1
2	2						2
3	4						4
4			1			1	1
5	2						2
6	2						2
7	3						3
8	1						1
9	1	1	2		3	5	6
10	1						1
11	1						1
12	1						1
13	1				1	1	2
14	3						3
15					1	1	1
16					1	1	1
17	2						2
18	2				1	1	3
19	3						3
20	1				1	1	2
21	1						1
22	2				1	1	3

ตารางที่ 16 (ต่อ)

ข้อความ	สกัดได้		สกัดความสัมพันธ์ไม่ได้			รวมความสัมพันธ์ที่สกัดไม่ได้	ความสัมพันธ์ทั้งหมดในข้อความ
	ความสัมพันธ์ที่สกัดได้ถูก	สกัดได้แต่ผิด	ปัญหาจากการสกัด NE	ปัญหาจากการตัดคำ	ไม่เข้าใจความหมายหรือผิดพลาด		
23	1	1	1			1	2
24	3						3
25	5						5
26	2						2
27	2						2
28	2						2
29	1						1
30	3		1			1	4
31	2						2
32					1	1	1
33	1						1
34	2						2
35	2						2
36	3		1			1	4
37	1				2	2	3
38	1						1
39	1						1
40	1						1
41	1						1
42	1				2	2	3
43	2						2
44				1		1	1

ตารางที่ 16 (ต่อ)

ข้อความ	สกัดได้		สกัดความสัมพันธ์ไม่ได้			รวม ความสัมพันธ์ที่สกัดไม่ได้	ความสัมพันธ์ทั้งหมดในข้อความ
	ความสัมพันธ์ที่สกัดได้ถูก	สกัดได้แต่ผิด	ปัญหาจากการสกัด NE	ปัญหาจากการตัดคำ	ไม่เข้าใจความหมายหรือผิดพลาด		
45	2						2
46	2						2
47	3						3
48	2	1					2
49	3						3
50	3	1					3
51				1		1	1
52	4						4
53	2			1		1	3
54			1		2	3	3
55	2						2
56	1						1
57	3						3
58	1		1			1	2
59	2						2
60	4						4
61	1						1
62	1		1		1	2	3
63	1						1
64	2						2
65	2						2
66	2	1					2

ตารางที่ 16 (ต่อ)

ข้อความ	สกัดได้		สกัดความสัมพันธ์ไม่ได้			รวมความสัมพันธ์ที่สกัดไม่ได้	ความสัมพันธ์ทั้งหมดในข้อความ
	ความสัมพันธ์ที่สกัดได้ถูก	สกัดได้แต่ผิด	ปัญหาจากการสกัด NE	ปัญหาจากการตัดคำ	ไม่เข้าใจความหมายหรือผิดพลาด		
67	1				1	1	2
68	2	1					2
69	1				1	1	2
70	2						2
71	3						3
72	1						1
73	1						1
74	1				1	1	2
75	3						3
76	1				1	1	2
77	2	1					2
78	1						1
79	2						2
80	1						1
81	2		1			1	3
82	4						4
83	1						1
84	3	1					3
85	1						1
86	1		1	2		3	4
87	2						2
88	2						2

ตารางที่ 16 (ต่อ)

ข้อความ	สกัดได้		สกัดความสัมพันธ์ไม่ได้			รวมความสัมพันธ์ที่สกัดไม่ได้	ความสัมพันธ์ทั้งหมดในข้อความ
	ความสัมพันธ์ที่สกัดได้ถูก	สกัดได้แต่ผิด	ปัญหาจากการสกัด NE	ปัญหาจากการตัดคำ	ไม่เข้าใจความหมายหรือผิดพลาด		
89	2				1	1	3
90	2	1					2
91	1						1
92	2						2
93	1						1
94	2				1	1	3
95	2						2
96	2		1			1	3
97	2	1					2
98	2						2
99	1						1
100	1						1
รวม	**172**	**10**	**12**	**5**	**23**	**40**	**212**
ความสัมพันธ์ที่ระบบเลือกมาทั้งหมด							**182**
ความสัมพันธ์ที่ถูกต้องทั้งหมดในข้อความ(ไม่รวมปัญหาตัดคำ และ NE)							**195**

ตารางที่ 17 ผลการสกัดความสัมพันธ์ในชุดทดสอบจริง (Training Set)

ข้อความ	สกัดได้		สกัดความสัมพันธ์ไม่ได้			รวมความสัมพันธ์ที่สกัดไม่ได้	ความสัมพันธ์ทั้งหมดในข้อความ
	ความสัมพันธ์ที่สกัดได้ถูก	สกัดได้แต่ผิด	ปัญหาจากการสกัด NE	ปัญหาจากการตัดคำ	ไม่เข้าใจความหมายหรือผิดพลาด		
1	1						1
2			1		1	2	2
3	3	1	1			1	4
4					1	1	1
5	1		1			1	2
6	2	1					2
7	2		1			1	3
8	4		1			1	5
9	1		1		1	1	2
10	1						1
11	1		1			1	2
12	1						1
13	1		2			3	4
14	1						1
15	3						3
16	1						1
17	1						1
18	2				1	1	3
19	2						2
20			1			1	1
21	3		1			1	4
22	1		1			1	2

ตารางที่ 17 (ต่อ)

ข้อความ	สกัดได้		สกัดความสัมพันธ์ไม่ได้			รวมความสัมพันธ์ที่สกัดไม่ได้	ความสัมพันธ์ทั้งหมดในข้อความ
	ความสัมพันธ์ที่สกัดได้ถูก	สกัดได้แต่ผิด	ปัญหาจากการสกัด NE	ปัญหาจากการตัดคำ	ไม่เข้าใจความหมายหรือผิดพลาด		
23			2			2	2
24	1		1			1	2
25	2						2
26	3		1			1	4
27	2	1	2			2	4
28	1	1			1	1	2
29	1						1
30	3						3
31	2				1	1	3
32							0
33	4						4
34	1		1			1	2
35		1	1			1	1
36			1			1	1
37	2		1		1	2	4
38	1		1			1	2
39	1						1
40	1				1	1	2
41			1			1	1
42	1		4			4	5
43				1		1	1
44	1						1

ตารางที่ 17 (ต่อ)

ข้อความ	สกัดได้		สกัดความสัมพันธ์ไม่ได้			รวมความสัมพันธ์ที่สกัดไม่ได้	ความสัมพันธ์ทั้งหมดในข้อความ
	ความสัมพันธ์ที่สกัดได้ถูก	สกัดได้แต่ผิด	ปัญหาจากการสกัด NE	ปัญหาจากการตัดคำ	ไม่เข้าใจความหมายหรือผิดพลาด		
45	1						1
46	2						2
47	2						2
48	2		1			1	3
49	3		1			1	4
50	2		1		1	2	4
51	1	1					1
52	2						2
53	1						1
54	2						2
55			2			2	2
56	2		1			1	3
57	5		2			2	7
58	2						2
59	2		1			1	3
60	1						1
61	1	1					1
62			2			2	2
63	1		2			2	3
64			2			2	2
65	1						1
66	1						1

ตารางที่ 17 (ต่อ)

ข้อความ	สกัดได้		สกัดความสัมพันธ์ไม่ได้			รวมความสัมพันธ์ที่สกัดไม่ได้	ความสัมพันธ์ทั้งหมดในข้อความ
	ความสัมพันธ์ที่สกัดได้ถูก	สกัดได้แต่ผิด	ปัญหาจากการสกัด NE	ปัญหาจากการตัดคำ	ไม่เข้าใจความหมายหรือผิดพลาด		
67	1		2	1	2	5	6
68	1		1	1	1	3	4
69	2				1	1	3
70	1						1
71	1	2	1			1	2
72	1						1
73	1		1			1	2
74	2						2
75	1						1
76	2						2
77	2		1			1	3
78	1						1
79	2						2
80	1	1	1			1	2
81					1	1	1
82	2		1			1	3
83	1		1			1	2
84	1						1
85	1		1		1	2	3
86	1						1
87	2						2
88	1		1			1	2

ตารางที่ 17 (ต่อ)

ข้อความ	สกัดได้		สกัดความสัมพันธ์ไม่ได้			รวมความสัมพันธ์ที่สกัดไม่ได้	ความสัมพันธ์ทั้งหมดในข้อความ
	ความสัมพันธ์ที่สกัดได้ถูก	สกัดได้แต่ผิด	ปัญหาจากการสกัด NE	ปัญหาจากการตัดคำ	ไม่เข้าใจความหมายหรือผิดพลาด		
89	1						1
90			1			1	1
91	1						1
92	2				1	1	3
93	1						1
94	1						1
95	1				2	2	3
96	1		1			1	2
97	1						1
98			1		1	2	2
99	2						2
100	2						2
101	1						1
102	1		1			1	2
103	1		1			1	2
104	1						1
105	1		1			1	2
106			1			1	1
107	1		2			2	3
108	1						1
109	2						2
110	1		3			3	4
111	1						1

ตารางที่ 17 (ต่อ)

ข้อความ	สกัดได้		สกัดความสัมพันธ์ไม่ได้			รวม	ความสัมพันธ์
	ความสัมพันธ์ที่สกัดได้ถูก	สกัดได้แต่ผิด	ปัญหาจากการสกัด NE	ปัญหาจากการตัดคำ	ไม่เข้าใจความหมายหรือผิดพลาด	ความสัมพันธ์ที่สกัดไม่ได้	ทั้งหมดในข้อความ
112	1						1
113	2						2
114	1						1
115	2				1	1	3
116	1	1			1	1	2
117	2		2		1	3	5
118			3		1	4	4
119	2		3			3	5
120	1		1			1	2
121			1		2	3	3
122	2	1					2
123	1				4	4	5
124	1						1
125			3			3	3
126			1			1	1
127			1			1	1
128	1						1
129	1						1
130	1		1			1	2
131	1						1
132	1						1
133	2	1	1		2	3	5
134	1		2		1	3	4

ตารางที่ 17 (ต่อ)

ข้อความ	สกัดได้		สกัดความสัมพันธ์ไม่ได้			รวมความสัมพันธ์ที่สกัดไม่ได้	ความสัมพันธ์ทั้งหมดในข้อความ
	ความสัมพันธ์ที่สกัดได้ถูก	สกัดได้แต่ผิด	ปัญหาจากการสกัด NE	ปัญหาจากการตัดคำ	ไม่เข้าใจความหมายหรือผิดพลาด		
135	2				1	1	3
136	1	1			2	2	3
137							0
138	2	1					2
139	2						2
140					1	1	1
141	2	1					2
142	2	1					2
143	1				1	1	2
144	1						1
145	2						2
146	3						3
147	1						1
148	1	3					1
149	2	1			1	1	3
150	1						1
151	1						1
152			1			1	1
153	1						1
154	1				1	1	2
155	3						3
156	1	2					1
157	2				1	1	3

ตารางที่ 17 (ต่อ)

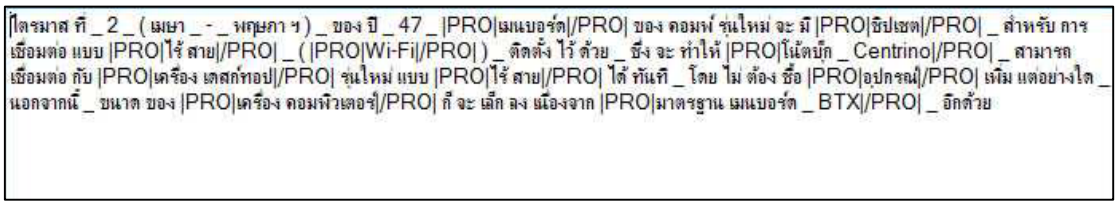
ข้อความ	สกัดได้		สกัดความสัมพันธ์ไม่ได้			รวมความสัมพันธ์ที่สกัดไม่ได้	ความสัมพันธ์ทั้งหมดในข้อความ
	ความสัมพันธ์ที่สกัดได้ถูก	สกัดได้แต่ผิด	ปัญหาจากการสกัด NE	ปัญหาจากการตัดคำ	ไม่เข้าใจความหมายหรือผิดพลาด		
158			1			1	1
159	1				2	2	3
160			2			2	2
161	3						3
162	3						3
163	1						1
164	1				1	1	2
165		1	1		1	2	2
166	1				1	1	2
167					1	1	1
168	2						2
169		1			1	1	1
170	4						4
171	1		1			1	2
172					1	1	1
173	1						1
174	1						1
175	2		1				2
176	1			1		1	2
177	1	1			2	2	3
178	2						2
179		1			1	1	1
180			2			2	2

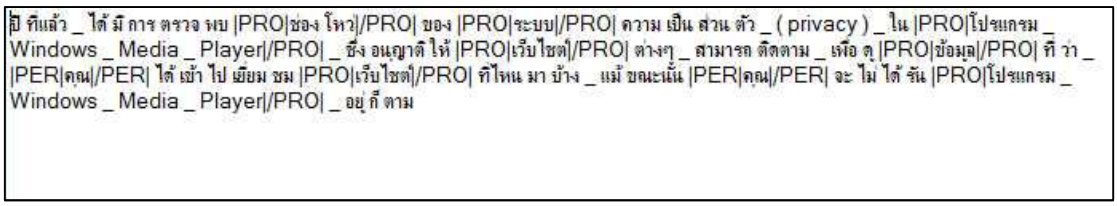
ตารางที่ 17 (ต่อ)

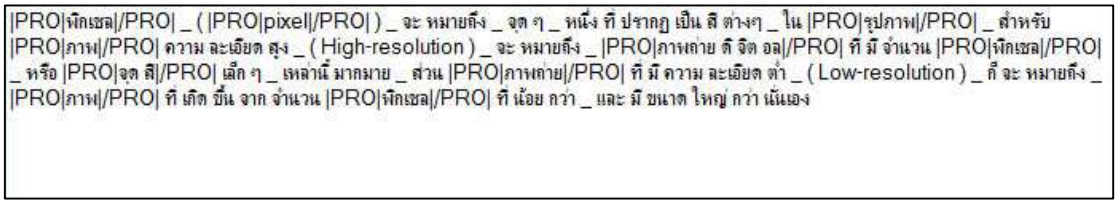
ข้อความ	สกัดได้		สกัดความสัมพันธ์ไม่ได้			รวมความสัมพันธ์ที่สกัดไม่ได้	ความสัมพันธ์ทั้งหมดในข้อความ
	ความสัมพันธ์ที่สกัดได้ถูก	สกัดได้แต่ผิด	ปัญหาจากการสกัด NE	ปัญหาจากการตัดคำ	ไม่เข้าใจความหมายหรือผิดพลาด		
181			1			1	1
182	2						2
183	2	1					2
184	2						2
185	2				2	2	4
186	1						1
187	1						1
188		1	1			1	1
189					1	1	1
190	1						1
191	1		1			1	2
192	1						1
193	1		1				1
194	2						2
195	1		2			2	3
196	1	1			1	1	2
197			1			1	1
198					1	1	1
199			1			1	1

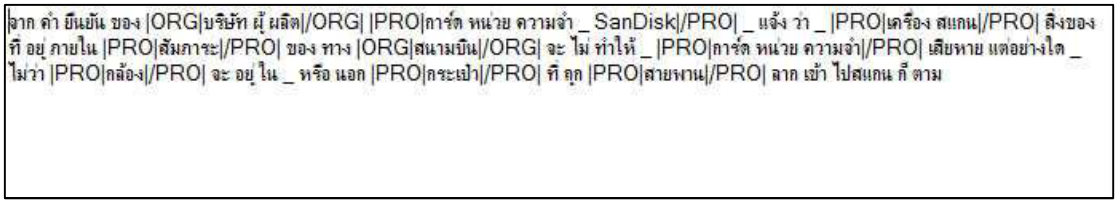
ตารางที่ 17 (ต่อ)

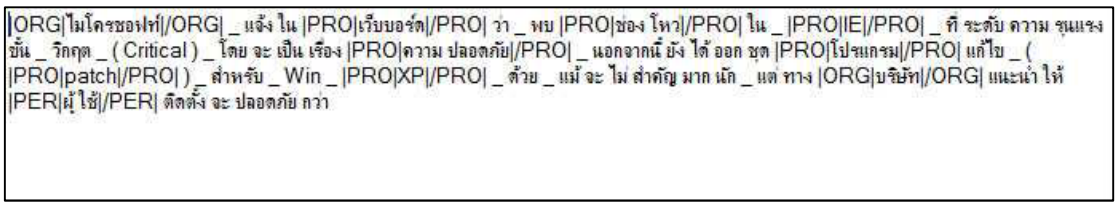
ข้อความ	สกัดได้		สกัดความสัมพันธ์ไม่ได้			รวมความสัมพันธ์ที่สกัดไม่ได้	ความสัมพันธ์ทั้งหมดในข้อความ
	ความสัมพันธ์ที่สกัดได้ถูก	สกัดได้แต่ผิด	ปัญหาจากการสกัด NE	ปัญหาจากการตัดคำ	ไม่เข้าใจความหมายหรือผิดพลาด		
200			2			2	2
รวม	**244**	**30**	**104**	**4**	**56**	**162**	**406**
ความสัมพันธ์ที่ระบบเลือกมาทั้งหมด							**274**
ความสัมพันธ์ที่ถูกต้องทั้งหมดในข้อความ(ไม่รวมปัญหาตัดคำ และ NE)							**300**

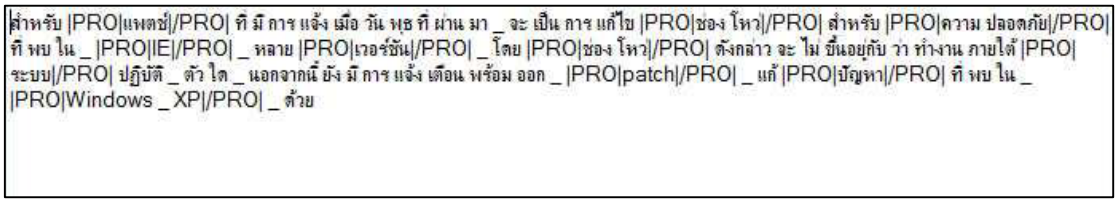


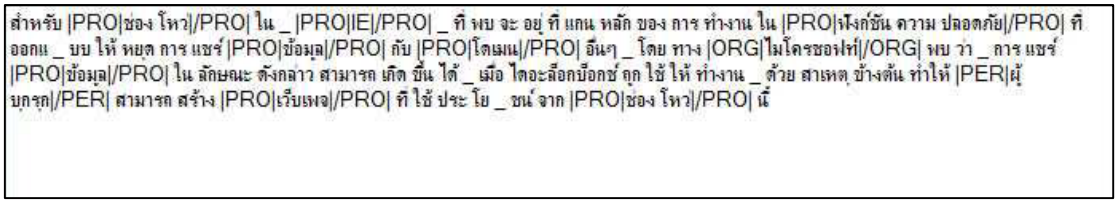












ภาพที่ 19 ตัวอย่างผลลัพธ์ของการสกัด NE

|PRO|คอลัมน์|/PRO| ตาวนโหลด วันนี้ _ ขอ แนะนำ โปรแกรมน่า สนใจ ชื่อ ว่า _ |PRO|SpywareBlaster|/PRO| _ ที่ จะ ช่วย ป้องกัน |PER|คุณ จาก|/PER| บรรดา _ |PRO|Spyware|/PRO| _ ต่างๆ _ ที่ พยายาม จะ แอบ ติดตั้ง ตัวเอง เข้า ไป ใน |PRO|ระบบ|/PRO| ของ |PER|คุณ|/PER| หน้า ที่ หลัก ของ _ |PRO|SpywareBlaster|/PRO| _ ก็ คือ _ ช่วย สอดส่อง ดูแล |PRO|ระบบ|/PRO| ให้ กับ |PER|คุณ|/PER| ตลอดเวลา

|PRO|SpywareBlaster|/PRO| _ จะ ตรวจ จับ _ |PRO|ActiveX _ controller|/PRO| _ ของ |PRO|สปายแวร์|/PRO| ที่ รู้จัก _ และ ป้องกัน การ ติดตั้ง คอนโทรล พวก นี้ จาก |PRO|เว็บเพจ|/PRO| เข้า สู่ |PRO|ระบบ|/PRO| ของ |PER|คุณ|/PER| _ หลังจาก ติดตั้ง _ |PRO|SpywareBlaster|/PRO| _ เข้า ไป แล้ว _ |PRO|สปายแวร์|/PRO| ยอด ฮิต อย่าง _ Gator _ ก็ จะ ไม่ มี โอกาส โผล่ หน้า มา ถาม |PER| คุณ|/PER| อีก เลย ว่า _ ต้องการ ตาวนโหลด ตัว มัน เข้า ไป ใน |PRO|ระบบ|/PRO| _ หรือไม่

|PRO|SpywareBlaster|/PRO| _ ยัง มี |PRO|ฟังก์ชัน _ System-restore|/PRO| _ ที่ คล้าย กับ ของ _ |PRO|Windows _ XP|/PRO| _ อีกด้วย _ โดย ใน การ ทำงาน |PRO|โปรแกรม|/PRO| จะ เก็บ สถานะ ของ |PRO|ระบบ|/PRO| ที่ ปราศจาก |PRO|สปายแวร์|/PRO| เอาไว้ _ และ เมื่อใด ก็ ตามที่ |PER|คุณ|/PER| เผลอ ติด |PRO|สปายแวร์|/PRO| เข้า ไป ใน |PRO|ระบบ|/PRO| _ (เช่น _ อาจ จะ ลืม อัพเดต |PRO|ฐาน ข้อมูล|/PRO|) _ |PER|คุณ|/PER| ก็ สามารถ เรียก |PRO|ระบบ|/PRO| กลับคืน สู่ สภาพ ก่อน ที่ จะ ติด |PRO|สปายแวร์|/PRO| ได้ เอ็นเอง

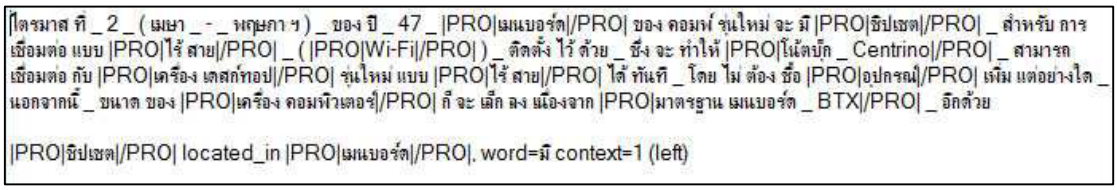
|PER|คุณ|/PER| จะ ต้อง เตรียม |PRO|กระดาษ พิมพ์|/PRO| นามบัตร ขนาด _ 2 _ x _ 3.5 _ นิ้ว _ ซึ่ง อาจ จะ ใช้ |PRO|กระดาษ _ A4|/PRO| _ ที่ แข็ง สักหน่อย ก็ ได้ _ จากนั้น เข้า ไป ที่ |PRO|เว็บไซต์ _ Microsoft|/PRO| _ เพื่อ ตาวนโหลด เทมเพลตฟรี _ สำหรับ ออกแบบ นามบัตร _ เลือก |PRO|เทมเพลต|/PRO| ที่ ถูกใจ _ จากนั้น แก้ไข รายละเอียด ตามที่ ต้องการ _ จัด วาง ตำแหน่ง _ แล้ว สั่ง พิมพ์ ออก มา ก็ เป็นอัน เรียบร้อย

ให้ |PER|คุณ ปิด|/PER| |PRO|โปรแกรม _ Word|/PRO| _ ก่อน _ จากนั้น คลิ๊ก |PRO|ปุ่ม _ Start|/PRO| _ เลือก |PRO|คำสั่ง _ Search|/PRO| _ / _ For _ Files _ or _ Folders _ คลิก เลือก |PRO|คำสั่ง _ All _ files _ and _ folders.|/PRO| _ พิมพ์ _ normal.dot _ เข้า ไป ใน ช่อง _ All _ or _ part _ of _ the _ filename: _ เมื่อ พบ |PRO|ไฟล์|/PRO| แล้ว ให้ คลิ๊ก เลือก _ กด |PRO|ปุ่ม _ F2|/PRO| _ เพื่อ กำหนด ชื่อ |PRO|ไฟล์|/PRO| เป็น ชื่อ อื่น เช่น _ abnormal.dot _ เปิด _ |PRO|Word|/PRO| _ ขึ้น ทำงาน _ |PRO|โปรแกรม|/PRO| จะ สร้าง |PRO|ไฟล์ _ normal.dot|/PRO| _ ขึ้น มา ใหม่ _ พร้อมทั้ง กำหนด ให้ เป็น ค่า ดีฟอลต์ โดย อัตโนมัติ

|ORG|สำนัก|/ORG| |PRO|ข่าว|/PRO| เอพี |PRO|รายงาน|/PRO| จาก เรดมอนด์ ว่า _ |ORG|ไมโครซอฟท์|/ORG| กำลัง จะ ออก ชุด |PRO| ซอฟต์แวร์|/PRO| แก้ไข _ (|PRO|patch|/PRO|) _ ตัว ใหม่ สำหรับ _ |PRO|ซอฟต์แวร์ _ Internet _ Explorer|/PRO| _ (|PRO|IE|/PRO|) _ เมื่องจาก _ |PRO|patch|/PRO| _ ตัว ล่าสุด _ ทำให้ |PRO|ฟังก์ชัน|/PRO| ที่ อนุญาต ให้ |PER|ผู้ ใช้|/PER| สามารถ เข้า ไป ใน |PRO|เว็บ ไซต์|/PRO| ที่ ได้ ลง |PRO|ทะเบียน|/PRO| ไว้ ก่อนหน้า นี้ เสียหาย

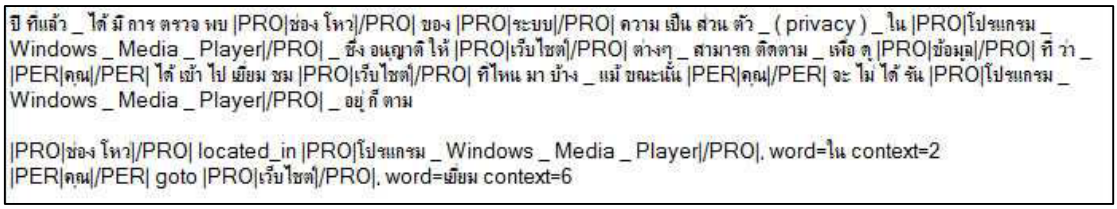
|ORG|ไมโครซอฟท์|/ORG| ได้ ออก ชุด |PRO|ซอฟต์แวร์|/PRO| แก้ไข |PRO|ปัญหา ช่อง โหว่|/PRO| ของ |PRO|ความ ปลอดภัย|/PRO| _ ที่ เปิด ช่องทาง ให้ แฮกเกอร์ สามารถ เข้าถึง |PRO|ข้อมูล|/PRO| ส่วน |PER|บุคคล|/PER| _ หรือ ควบคุม การ ทำงาน |PRO|พีซี|/PRO| ของ |PER|ผู้ ใช้|/PER| _ |PRO|IE _ เวอร์ชัน _ 5.01, _ 5.5|/PRO| _ และ _ 6.0 _ แต่ หลังจาก ได้ มี การ เปิด ให้ ตาวนโหลด ชุด |PRO| ซอฟต์แวร์|/PRO| แก้ไข ตัว ล่าสุด _ ออก ไป แล้ว _ |PER|ผู้ ใช้|/PER| หลาย ราย ได้ ติดต่อ เข้า ไป ยัง |ORG|ไมโครซอฟท์|/ORG|

ภาพที่ 19 (ต่อ)



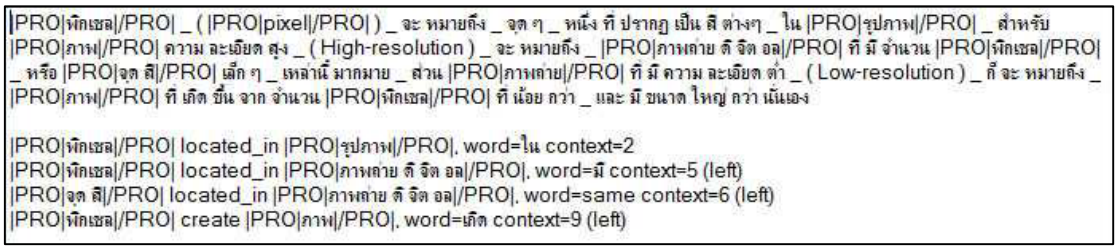
```
ไตรมาส ที่ _ 2 _ ( เมษา _ - _ พฤษภา ฯ ) _ ของ ปี _ 47 _ |PRO|เมนบอร์ด|/PRO| ของ คอมพ์ รุ่นใหม่ จะ มี |PRO|ชิปเซต|/PRO| _ สำหรับ การ
เชื่อมต่อ แบบ |PRO|ไร้ สาย|/PRO| _ ( |PRO|Wi-Fi|/PRO| ) _ ติดตั้ง ไว้ ด้วย _ ซึ่ง จะ ทำให้ |PRO|โน้ตบุ๊ก _ Centrino|/PRO| _ สามารถ
เชื่อมต่อ กับ |PRO|เครือง เดสก์ทอป|/PRO| รุ่นใหม่ แบบ |PRO|ไร้ สาย|/PRO| ได้ ทันที _ โดย ไม่ ต้อง ซื้อ |PRO|อุปกรณ์|/PRO| เพิ่ม แต่อย่างใด _
นอกจากนี้ _ ขนาด ของ |PRO|เครื่อง คอมพิวเตอร์|/PRO| ก็ จะ เล็ก ลง เนื่องจาก |PRO|มาตรฐาน เมนบอร์ด _ BTX|/PRO| _ อีกด้วย

|PRO|ชิปเซต|/PRO| located_in |PRO|เมนบอร์ด|/PRO|. word=มี context=1 (left)
```



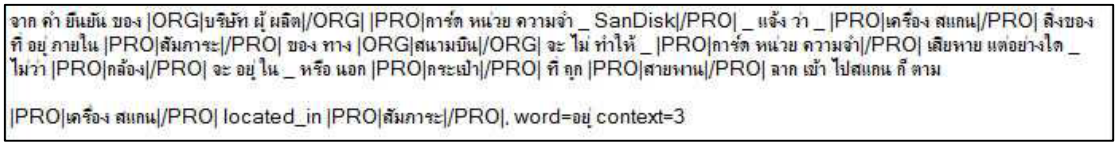
```
ปี ที่แล้ว _ ได้ มี การ ตรวจ พบ |PRO|ช่อง โหว่|/PRO| ของ |PRO|ระบบ|/PRO| ความ เป็น ส่วน ตัว _ ( privacy ) _ ใน |PRO|โปรแกรม _
Windows _ Media _ Player|/PRO| _ ซึ่ง อนุญาต ให้ |PRO|เว็บไซต์|/PRO| ต่างๆ _ สามารถ ติดตาม _ เพื่อ ดู |PRO|ข้อมูล|/PRO| ที่ ว่า _
|PER|คุณ|/PER| ได้ เข้า ไป เยี่ยม ชม |PRO|เว็บไซต์|/PRO| ที่ไหน มา บ้าง _ แม้ ขณะนั้น |PER|คุณ|/PER| จะ ไม่ ได้ รัน |PRO|โปรแกรม _
Windows _ Media _ Player|/PRO| _ อยู่ ก็ ตาม

|PRO|ช่อง โหว่|/PRO| located_in |PRO|โปรแกรม _ Windows _ Media _ Player|/PRO|. word=ใน context=2
|PER|คุณ|/PER| goto |PRO|เว็บไซต์|/PRO|. word=เยี่ยม context=6
```



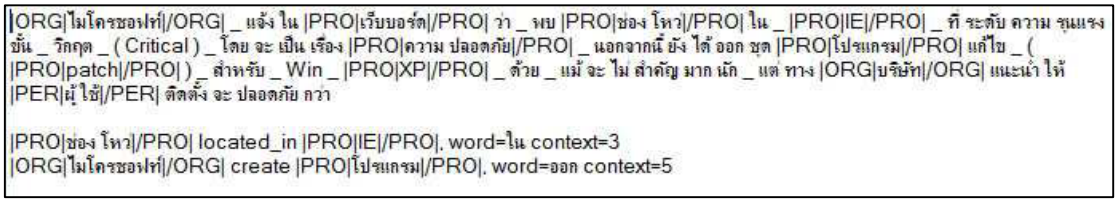
```
|PRO|พิกเซล|/PRO| _ ( |PRO|pixel|/PRO| ) _ จะ หมายถึง _ จุด ๆ _ หนึ่ง ที่ ปรากฏ เป็น สี ต่างๆ _ ใน |PRO|รูปภาพ|/PRO| _ สำหรับ
|PRO|ภาพ|/PRO| ความ ละเอียด สูง _ ( High-resolution ) _ จะ หมายถึง _ |PRO|ภาพถ่าย ดิ จิต อล|/PRO| ที่ มี จำนวน |PRO|พิกเซล|/PRO|
_ หรือ |PRO|จุด สี|/PRO| เล็ก ๆ _ เหล่านี้ มากมาย _ ส่วน |PRO|ภาพถ่าย|/PRO| ที่ มี ความ ละเอียด ต่ำ _ ( Low-resolution ) _ ก็ จะ หมายถึง _
|PRO|ภาพ|/PRO| ที่ เกิด ขึ้น จาก จำนวน |PRO|พิกเซล|/PRO| ที่ น้อย กว่า _ และ มี ขนาด ใหญ่ กว่า เนื้อง

|PRO|พิกเซล|/PRO| located_in |PRO|รูปภาพ|/PRO|. word=ใน context=2
|PRO|พิกเซล|/PRO| located_in |PRO|ภาพถ่าย ดิ จิต อล|/PRO|. word=มี context=5 (left)
|PRO|จุด สี|/PRO| located_in |PRO|ภาพถ่าย ดิ จิต อล|/PRO|. word=same context=6 (left)
|PRO|พิกเซล|/PRO| create |PRO|ภาพ|/PRO|. word=เกิด context=9 (left)
```



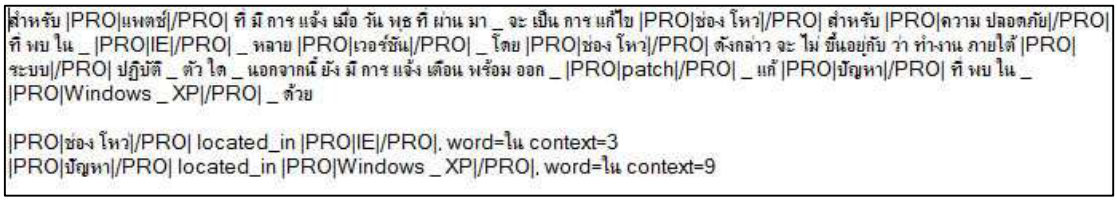
```
จาก คำ ยืนยัน ของ |ORG|บริษัท ผู้ ผลิต|/ORG| |PRO|การ์ด หน่วย ความจำ _ SanDisk|/PRO| _ แจ้ง ว่า _ |PRO|เครื่อง สแกน|/PRO| สิ่งของ
ที่ อยู่ ภายใน |PRO|สัมภาระ|/PRO| ของ ทาง |ORG|สนามบิน|/ORG| จะ ไม่ ทำให้ _ |PRO|การ์ด หน่วย ความจำ|/PRO| เสียหาย แต่อย่างใด _
ไม่ว่า |PRO|กล้อง|/PRO| จะ อยู่ ใน _ หรือ นอก |PRO|กระเป๋า|/PRO| ที่ ถูก |PRO|สายพาน|/PRO| ลาก เข้า ไปสแกน ก็ ตาม

|PRO|เครื่อง สแกน|/PRO| located_in |PRO|สัมภาระ|/PRO|. word=อยู่ context=3
```



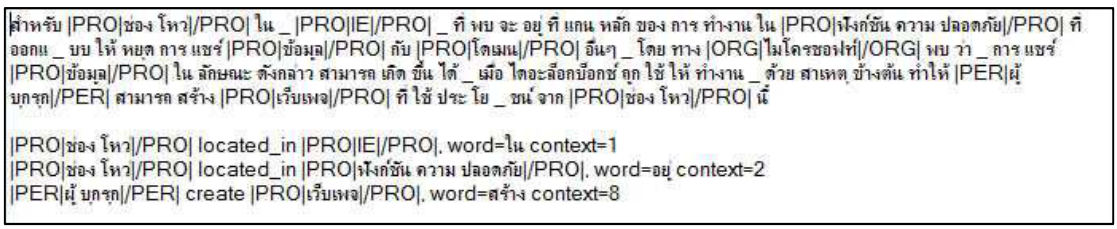
```
|ORG|ไมโครซอฟท์|/ORG| _ แจ้ง ใน |PRO|เว็บบอร์ด|/PRO| ว่า _ พบ |PRO|ช่อง โหว่|/PRO| ใน _ |PRO|IE|/PRO| _ ที่ ระดับ ความ รุนแรง
ขั้น _ วิกฤต _ ( Critical ) _ โดย จะ เป็น เรื่อง |PRO|ความ ปลอดภัย|/PRO| _ นอกจากนี้ ยัง ได้ ออก ชุด |PRO|โปรแกรม|/PRO| แก้ไข _ (
|PRO|patch|/PRO| ) _ สำหรับ _ Win _ |PRO|XP|/PRO| _ ด้วย _ แม้ จะ ไม่ สำคัญ มาก นัก _ แต่ ทาง |ORG|บริษัท|/ORG| แนะนำ ให้
|PER|ผู้ ใช้|/PER| ติดตั้ง จะ ปลอดภัย กว่า

|PRO|ช่อง โหว่|/PRO| located_in |PRO|IE|/PRO|. word=ใน context=3
|ORG|ไมโครซอฟท์|/ORG| create |PRO|โปรแกรม|/PRO|. word=ออก context=5
```



```
สำหรับ |PRO|แพตช์|/PRO| ที่ มี การ แจ้ง เมื่อ วัน พฤ ที่ ผ่าน มา _ จะ เป็น การ แก้ไข |PRO|ช่อง โหว่|/PRO| สำหรับ |PRO|ความ ปลอดภัย|/PRO|
ที่ พบ ใน _ |PRO|IE|/PRO| _ หลาย |PRO|เวอร์ชัน|/PRO| _ โดย |PRO|ช่อง โหว่|/PRO| ดังกล่าว จะ ไม่ ขึ้นอยู่กับ ว่า ทำงาน ภายใต้ |PRO|
ระบบ|/PRO| ปฏิบัติ _ ตัว ใด _ นอกจากนี้ ยัง มี การ แจ้ง เตือน พร้อม ออก _ |PRO|patch|/PRO| _ แก้ |PRO|ปัญหา|/PRO| ที่ พบ ใน _
|PRO|Windows _ XP|/PRO| _ ด้วย

|PRO|ช่อง โหว่|/PRO| located_in |PRO|IE|/PRO|. word=ใน context=3
|PRO|ปัญหา|/PRO| located_in |PRO|Windows _ XP|/PRO|. word=ใน context=9
```



```
สำหรับ |PRO|ช่อง โหว่|/PRO| ใน _ |PRO|IE|/PRO| _ ที่ พบ จะ อยู่ ที่ แกน หลัก ของ การ ทำงาน ใน |PRO|ฟังก์ชัน ความ ปลอดภัย|/PRO| ที่
ออกแ _ บบ ให้ หยุด การ แชร์ |PRO|ข้อมูล|/PRO| กับ |PRO|โดเมน|/PRO| อื่นๆ _ โดย ทาง |ORG|ไมโครซอฟท์|/ORG| พบ ว่า _ การ แชร์
|PRO|ข้อมูล|/PRO| ใน ลักษณะ ดังกล่าว สามารถ เกิด ขึ้น ได้ _ เมื่อ ไดอะล็อกบ็อกซ์ ถูก ใช้ ให้ ทำงาน _ ด้วย สาเหตุ ข้างต้น ทำให้ |PER|ผู้
บุกรุก|/PER| สามารถ สร้าง |PRO|เว็บเพจ|/PRO| ที่ ใช้ ประ โย _ ชน์ จาก |PRO|ช่อง โหว่|/PRO| นี้

|PRO|ช่อง โหว่|/PRO| located_in |PRO|IE|/PRO|. word=ใน context=1
|PRO|ช่อง โหว่|/PRO| located_in |PRO|ฟังก์ชัน ความ ปลอดภัย|/PRO|. word=อยู่ context=2
|PER|ผู้ บุกรุก|/PER| create |PRO|เว็บเพจ|/PRO|. word=สร้าง context=8
```

ภาพที่ 20 ตัวอย่างผลลัพธ์ของการสกัดความสัมพันธ์

|PRO|คอลัมน์|/PRO| ตาวนโหลด วันนี้ _ ขอ แนะนำ โปรแกรมน่า สนใจ ชื่อ ว่า _ |PRO|SpywareBlaster|/PRO| _ ที่ จะ ช่วย ป้องกัน |PER|คุณ จาก|/PER| บรรดา _ |PRO|Spyware|/PRO| _ ต่างๆ _ ที่ พยายาม จะ แอบ ติดตั้ง ตัวเอง เข้า ไป ใน |PRO|ระบบ|/PRO| ของ |PER|คุณ|/PER| หน้า ที่ หลัก ของ _ |PRO|SpywareBlaster|/PRO| _ ก็ คือ _ ช่วย สอดส่อง ดูแล |PRO|ระบบ|/PRO| ให้ กับ |PER|คุณ|/PER| ตลอดเวลา

|PRO|Spyware|/PRO| goto |PRO|ระบบ|/PRO|. word=เข้า context=4

|PRO|SpywareBlaster|/PRO| _ จะ ตรวจ จับ _ |PRO|ActiveX _ controller|/PRO| _ ของ |PRO|สปายแวร์|/PRO| ที่ รู้จัก _ และ ป้องกัน การ ติดตั้ง คอนโทรล พวก นี้ จาก |PRO|เว็บเพจ|/PRO| เข้า สู่ |PRO|ระบบ|/PRO| ของ |PER|คุณ|/PER| _ หลังจาก ติดตั้ง _ |PRO|SpywareBlaster|/PRO| _ เข้า ไป แล้ว _ |PRO|สปายแวร์|/PRO| ยอด ฮิต อย่าง _ Gator _ ก็ จะ ไม่ มี โอกาส โผล่ หน้า มา ถาม |PER| คุณ|/PER| อีก เลย ว่า _ ต้องการ ตาวนโหลด ตัว มัน เข้า ไป ใน |PRO|ระบบ|/PRO| _ หรือไม่

|PRO|เว็บเพจ|/PRO| goto |PRO|ระบบ|/PRO|. word=เข้า context=4
|PRO|ระบบ|/PRO| located_in |PRO|SpywareBlaster|/PRO|. word=ติดตั้ง context=6
|PER|คุณ|/PER| goto |PRO|ระบบ|/PRO|. word=เข้า context=9

|PRO|SpywareBlaster|/PRO| _ ยัง มี |PRO|ฟังก์ชัน _ System-restore|/PRO| _ ที่ คล้าย กับ ของ _ |PRO|Windows _ XP|/PRO| _ อีกด้วย _ โดย ใน การ ทำงาน |PRO|โปรแกรม|/PRO| จะ เก็บ สถานะ ของ |PRO|ระบบ|/PRO| ที่ ปราศจาก |PRO|สปายแวร์|/PRO| เอาไว้ _ และ เมื่อใด ก็ ตามที่ |PER|คุณ|/PER| เผลอ ติด |PRO|สปายแวร์|/PRO| เข้า ไป ใน |PRO|ระบบ|/PRO| _ (เช่น _ อาจ จะ ลืม อัพเดต |PRO|ฐาน ข้อมูล|/PRO|) _ |PER|คุณ|/PER| ก็ สามารถ เรียก |PRO|ระบบ|/PRO| กลับคืน สู่ สภาพ ก่อน ที่ จะ ติด |PRO|สปายแวร์|/PRO| ได้ นั่นเอง

|PRO|ฟังก์ชัน _ System-restore|/PRO| located_in |PRO|SpywareBlaster|/PRO|. word=มี context=1 (left)
|PRO|โปรแกรม|/PRO| located_in |PRO|ระบบ|/PRO|. word=เก็บ context=4
|PRO|สปายแวร์|/PRO| located_in |PER|คุณ|/PER|. word=ติด context=7 (left)
|PRO|สปายแวร์|/PRO| goto |PRO|ระบบ|/PRO|. word=เข้า context=8
|PRO|สปายแวร์|/PRO| located_in |PRO|ระบบ|/PRO|. word=ติด context=12 (left)

|PER|คุณ|/PER| จะ ต้อง เตรียม |PRO|กระดาษ พิมพ์|/PRO| นามบัตร ขนาด _ 2 _ x _ 3.5 _ นิ้ว _ ซึ่ง อาจ จะ ใช้ |PRO|กระดาษ _ A4|/PRO| _ ที่ แข็ง สักหน่อย ก็ ได้ _ จากนั้น เข้า ไป ที่ |PRO|เว็บไซต์ _ Microsoft|/PRO| _ เพื่อ ตาวนโหลด เทมเพลตฟรี _ สำหรับ ออกแบบ นามบัตร _ เลือก |PRO|เทมเพลต|/PRO| ที่ ถูกใจ _ จากนั้น แก้ไข รายละเอียด ตามที่ ต้องการ _ จัด วาง ตำแหน่ง _ แล้ว สั่ง พิมพ์ ออก มา ก็ เป็นอัน เรียบร้อย

|PRO|กระดาษ _ A4|/PRO| goto |PRO|เว็บไซต์ _ Microsoft|/PRO|. word=ไป context=3

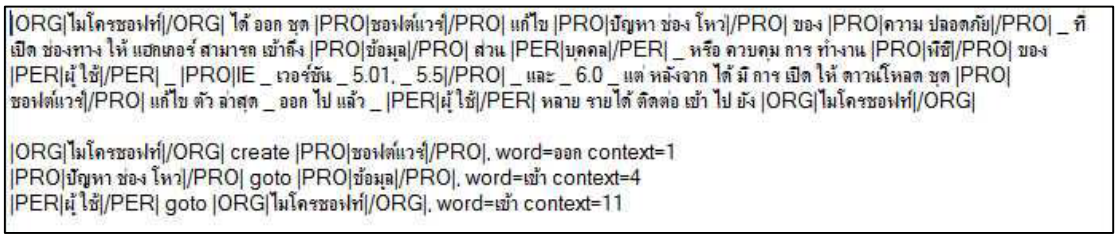
ให้ |PER|คุณ ปิด|/PER| |PRO|โปรแกรม _ Word|/PRO| _ ก่อน _ จากนั้น คลิก |PRO|ปุ่ม _ Start|/PRO| _ เลือก |PRO|คำสั่ง _ Search|/PRO| _ / _ For _ Files _ or _ Folders _ คลิก เลือก |PRO|คำสั่ง _ All _ files _ and _ folders.|/PRO| _ พิมพ์ _ normal.dot _ เข้า ไป ใน ช่อง _ All _ or _ part _ of _ the _ filename: _ เมื่อ จบ |PRO|ไฟล์|/PRO| แล้ว ให้ คลิก เลือก _ กด |PRO|ปุ่ม _ F2|/PRO| _ เพื่อ กำหนด ชื่อ |PRO|ไฟล์|/PRO| เป็น ชื่อ อื่น เช่น _ abnormal.dot _ เปิด _ |PRO|Word|/PRO| _ ขึ้น ทำงาน _ |PRO|โปรแกรม|/PRO| จะ สร้าง |PRO|ไฟล์ _ normal.dot|/PRO| _ ขึ้น มา ใหม่ _ พร้อมทั้ง กำหนด ให้ เป็น ค่า ดีฟอลต์ โดย อัตโนมัติ

|PRO|คำสั่ง _ All _ files _ and _ folders.|/PRO| create |PRO|ไฟล์|/PRO|. word=พิมพ์ context=5
|PRO|โปรแกรม|/PRO| create |PRO|ไฟล์ _ normal.dot|/PRO|. word=สร้าง context=10

|ORG|สำนัก|/ORG| |PRO|ข่าว|/PRO| เอพี |PRO|รายงาน|/PRO| จาก เรดมอนด์ ว่า _ |ORG|ไมโครซอฟท์|/ORG| กำลัง จะ ออก ชุด |PRO| ซอฟต์แวร์|/PRO| แก้ไข _ (|PRO|patch|/PRO|) _ ตัว ใหม่ สำหรับ _ |PRO|ซอฟต์แวร์ _ Internet _ Explorer|/PRO| _ (|PRO|IE|/PRO|) _ เมื่องจาก _ |PRO|patch|/PRO| _ ตัว ล่าสุด _ ทำให้ |PRO|ฟังก์ชัน|/PRO| ที่ อนุญาต ให้ |PER|ผู้ ใช้|/PER| สามารถ เข้า ไป ใน |PRO|เว็บ ไซต์|/PRO| ที่ ได้ ลง |PRO|ทะเบียน|/PRO| ไว้ ก่อนหน้า นี้ เสียหาย

|ORG|ไมโครซอฟท์|/ORG| create |PRO|ซอฟต์แวร์|/PRO|. word=ออก context=4
|PER|ผู้ ใช้|/PER| goto |PRO|เว็บไซต์|/PRO|. word=เข้า context=11
|PRO|เว็บไซต์|/PRO| goto |PRO|ทะเบียน|/PRO|. word=ลง context=12

ภาพที่ 20 (ต่อ)


|ORG|ไมโครซอฟท์|/ORG| ได้ ออก ชุด |PRO|ซอฟต์แวร์|/PRO| แก้ไข |PRO|ปัญหา ช่อง โหว่|/PRO| ของ |PRO|ความ ปลอดภัย|/PRO| _ ที่ เปิด ช่องทาง ให้ แฮกเกอร์ สามารถ เข้าถึง |PRO|ข้อมูล|/PRO| ส่วน |PER|บุคคล|/PER| _ หรือ ควบคุม การ ทำงาน |PRO|พีซี|/PRO| ของ |PER|ผู้ ใช้|/PER| _ |PRO|IE _ เวอร์ชัน _ 5.01, _ 5.5|/PRO| _ และ _ 6.0 _ แต่ หลังจาก ได้ มี การ เปิด ให้ ดาวน์โหลด ชุด |PRO| ซอฟต์แวร์|/PRO| แก้ไข ตัว ล่าสุด _ ออก ไป แล้ว _ |PER|ผู้ ใช้|/PER| หลาย ราย ได้ ติดต่อ เข้า ไป ยัง |ORG|ไมโครซอฟท์|/ORG|

|ORG|ไมโครซอฟท์|/ORG| create |PRO|ซอฟต์แวร์|/PRO|, word=ออก context=1
|PRO|ปัญหา ช่อง โหว่|/PRO| goto |PRO|ข้อมูล|/PRO|, word=เข้า context=4
|PER|ผู้ ใช้|/PER| goto |ORG|ไมโครซอฟท์|/ORG|, word=เข้า context=11

ภาพที่ 20 (ต่อ)

ประวัติผู้วิจัย

ชื่อ-สกุล	นายรัฐภูมิ ตันสุตะพานิช
ที่อยู่	201 หมู่ 4 ซ.รามอินทรา 5 แยก 15 ถนนรามอินทรา แขวงอนุสาวรีย์ เขตบางเขน กรุงเทพฯ 10220
ประวัติการศึกษา	
พ.ศ. 2548	สำเร็จการศึกษาศึกษาศาสตร์บัณฑิต สาขาคอมพิวเตอร์ศึกษา คณะศึกษาศาสตร์ มหาวิทยาลัยเทคโนโลยีราชมงคลธัญบุรี
พ.ศ. 2548	ศึกษาต่อวิทยาศาสตร์มหาบัณฑิต สาขาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยศิลปากร
ประวัติการทำงาน	
ต.ค. 2548- ม.ค. 2551	เจ้าหน้าที่ระบบงานคอมพิวเตอร์ มหาวิทยาลัยหอการค้าไทย