ใบรับรองวิทยานิพนธ์

บัณฑิตวิทยาลัย มหาวิทยาลัยเกษตรศาสตร์

วิทยาศาสตรมหาบัณฑิต (วิทยาการคอมพิวเตอร์)

ปริญญา

วิทยาการคอมพิวเตอร์ | วิทยาการคอมพิวเตอร์

สาขา | **ภาควิชา**

เรื่อง การสกัดเอกสารอ้างอิงอัตโนมัติ

Automatic Citation Extraction

นามผู้วิจัย นายพิชัย กิตติคง

ได้พิจารณาเห็นชอบโดย

ประธานกรรมการ .....

(รองศาสตราจารย์ชุลีรัตน์ จรัสกุลชัย, D.Sc.)

กรรมการ .....

(ผู้ช่วยศาสตราจารย์วรเศรษฐ สุวรรณิก, วศ.ด.)

กรรมการ .....

(รองศาสตราจารย์ศิริพร อ่องรุ่งเรือง, M.S.)

หัวหน้าภาควิชา .....

(ผู้ช่วยศาสตราจารย์ศิริกร จันทร์นวล, M.S.)

บัณฑิตวิทยาลัย มหาวิทยาลัยเกษตรศาสตร์รับรองแล้ว

.....

(รองศาสตราจารย์กัญจนา ธีระกุล, D.Agr.)

คณบดีบัณฑิตวิทยาลัย

วันที่ ..... **เดือน** ..... **พ.ศ.** .....

วิทยานิพนธ์

เรื่อง

การสกัดเอกสารอ้างอิงอัตโนมัติ

Automatic Citation Extraction

โดย

นายพิชัย กิตติคง

เสนอ

บัณฑิตวิทยาลัย มหาวิทยาลัยเกษตรศาสตร์
เพื่อความสมบูรณ์แห่งปริญญาวิทยาศาสตรมหาบัณฑิต (วิทยาการคอมพิวเตอร์)
พ.ศ. 2553

พิชัย กิตติคง 2553: การสกัดเอกสารอ้างอิงอัตโนมัติ ปริญญาวิทยาศาสตรมหาบัณฑิต (วิทยาการคอมพิวเตอร์) สาขาวิทยาการคอมพิวเตอร์ ภาควิชาวิทยาการคอมพิวเตอร์ ประธานกรรมการที่ปรึกษา: รองศาสตราจารย์ชุลีรัตน์ จรัสกุลชัย, D.Sc 76 หน้า

งานวิจัยฉบับนี้นำเสนอระบบสกัดเอกสารอ้างอิงแบบอัตโนมัติ การสกัดเอกสารอ้างอิงนั้นทำได้ยาก เนื่องจากรูปแบบการเขียนเอกสารมีหลายรูปแบบ งานวิจัยนี้นำวิธีการแบบ Rule-based มาประยุกต์ใช้เพื่อสกัดเอกสารอ้างอิงจากวิทยานิพนธ์ โดยทำการเขียนรูปแบบการอ้างอิงภาษาไทยในรูปของไวยากรณ์ไม่พึ่งบริบท (Context Free Grammar) โดยทำการทดสอบกับเอกสารอ้างอิงภาษาไทยในวิทยานิพนธ์ภาษาไทยเท่านั้น ซึ่งสามารถสกัด ชื่อผู้แต่ง, ชื่อเรื่อง และปีที่พิมพ์ ได้อย่างแม่นยำ ทำการวัดประสิทธิภาพกับชุดข้อมูลทดสอบที่สกัดเอกสารอ้างอิงด้วยมือโดยเปรียบเทียบ ชื่อผู้แต่ง, ชื่อเรื่อง และ ปีที่พิมพ์ โดยมีค่าเฉลี่ยความถูกต้องในการสกัดทั้งหมดที่ 96.49% ของรายการเอกสารอ้างอิง

การทำดัชนีอ้างอิงงานวิจัยฉบับนี้นำเสนออัลกอริทึม Levenshtein distance ในการตรวจสอบความคล้ายกันของหัวเรื่องวิทยานิพนธ์ ชื่อผู้แต่ง ที่ถูกอ้างอิงในวิทยานิพนธ์อื่น ๆ และทำการทดสอบ อัลกอลิทึมกับฐานข้อมูลวิทยานิพนธ์ของมหาวิทยาลัยเกษตรศาสตร์ที่ตีพิมพ์ระหว่างปี 2545 – 2548 ซึ่งมีความถูกต้องในการทำดัชนีอ้างอิงที่ 95.90% โดยวัดประสิทธิภาพกับการทำดัชนีอ้างอิงด้วยมือ

_____ _____ ___ / ___ / ___

ลายมือชื่อนิสิต ลายมือชื่อประธานกรรมการ

Pichai Kittikong 2010: Automatic Citation Extraction. Master of Science (Computer Science), Major Field: Computer Science, Department of Computer Science. Thesis Advisor: Associate Professor Chuleerat Jaruskulchai, D.Sc. 76 pages.

This research presents an automatic Thai citation extraction. Automatic citation extraction is difficult due to variations reference styles and Thai script. Thai citation extraction adopts the framework of rule-based to perform parsing Thai citation data from thai thesis. The Thai reference style is encoded by Context Free Grammar (CFG). The Accuracy of this algorithm has been tested on Thai thesis collection and only Thai references are extracted. The system can extract author, title, year with a high accuracy. The accuracy is compare with maual extraction by author, title and year. The overall average field accuracy of citation extraction is 96.49% of references.

Levenshtien distance is proposed for matching thesis title and author name. This research is tested on Thai thesis database collected from Kasetsart University during 2002 - 2005. The overall average citation index accuracy is 95.90% compare by manual citation index.

_____ _____ ___ / ___ / ___

Student's signature Thesis Advisor's signature

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงได้ด้วยดี จากความช่วยเหลือจากอาจารย์ที่ปรึกษา รศ.ดร. ชุลีรัตน์ จรัสกุลชัย ประธานกรรมการ ที่ได้ให้คำแนะนำและคำปรึกษาในการทำงานวิจัย รวมทั้งขอขอบพระคุณ อาจารย์วรเศรษฐ สุวรรณิก กรรมการสาขาวิชาเอก และ รศ.ดร. ศิริพร อ่องรุ่งเรือง กรรมการสาขาวิชารอง ในการปรับปรุงแก้ไขวิทยานิพนธ์ฉบับนี้ รวมทั้งสละเวลาในการเป็นกรรมการในการสอบอีกด้วย

ขอกราบขอบคุณบิดา มารดา รวมทั้งคณาจารย์ พี่ น้อง และเพื่อนๆ ที่ให้ความช่วยเหลือในการทำวิทยานิพนธ์จนสำเร็จลุล่วงได้

ขอขอบคุณเจ้าหน้าที่ของภาควิทยาการคอมพิวเตอร์ทุกท่านที่อำนวยความสะดวกในการทำงานวิจัยและติดต่อดำเนินการทุกอย่าง

คุณค่าและประโยชน์จากวิทยานิพนธ์เล่มนี้ ขอมอบแด่ผู้มีพระคุณทุกท่านที่กล่าวถึง

พิชัย กิตติคง

มิถุนายน 2553

สารบัญ

หน้า

สารบัญตาราง

สารบัญภาพ

ภาพผนวกที่

การสกัดเอกสารอ้างอิงอัตโนมัติ

Automatic Citation Extraction

คำนำ

การสกัดข้อมูล (Information Extraction) เป็นกระบวนการดึงข้อมูลที่ผู้ใช้ให้ความสนใจและต้องการคัดแยกออกมาจากเอกสารที่ไม่มีรูปแบบ และจัดเก็บใหม่ในรูปแบบที่มีโครงสร้าง เพื่อสามารถนำข้อมูลไปใช้งานต่อได้ง่ายด้วยการประมวลผลโดยคอมพิวเตอร์ ในปัจจุบันข้อมูลมีปริมาณมากขึ้นและส่วนใหญ่เป็นข้อมูลไม่มีโครงสร้าง การสกัดเอกสารช่วยเพิ่มปริมาณข้อมูลสารสนเทศที่ใช้ประโยชน์ได้มากขึ้น ทำให้มีการนำหลักการสกัดข้อมูลไปใช้ในหลากหลายแขนง เช่น การสกัดชื่อ (Name Entity Recognition), การสกัดคำ (Terminology Extraction) หรือ แบ่งตามประเภทของข้อมูลที่นำมาสกัด เช่น การสกัดข้อมูล DNA (DNA Extraction), การสกัดเอกสารอ้างอิง (Citation Extraction) และอื่นๆ

รายการเอกสารอ้างอิงเป็นรายชื่อเอกสารที่ผู้วิจัยใช้ศึกษาค้นคว้าในการทำงานวิจัย โดยจะเขียนระบุไว้ตอนท้ายของงานวิจัย ซึ่งการเขียนเอกสารอ้างอิงมีวิธีการเขียนอยู่หลากหลายรูปแบบตามแต่ละมาตรฐาน โดยข้อมูลของเอกสารอ้างอิงแต่ละมาตรฐานจะมีชนิดข้อมูลที่คล้ายกันแตกต่างกันที่ลำดับและสัญลักษณ์ในการแบ่งส่วนประกอบแตกต่างกันไป การสกัดเอกสารอ้างอิงจึงเป็นการระบุชนิดข้อมูลของเอกสารอ้างอิงว่าในแต่ละส่วนประกอบเป็นชนิดข้อมูลอะไร

การเขียนเอกสารอ้างอิงมีประโยชน์มากในการทำงานวิจัยเพราะแสดงถึงความน่าเชื่อถือในเนื้อหาที่ได้กล่าวถึงว่ามีหลักฐานที่อ้างอิงได้ ทำให้ผู้ที่อ่านงานวิจัยสามารถศึกษาค้นคว้าเพิ่มเติมได้อย่างสะดวก และอีกทั้งยังเป็นการให้เกียรติแก่เจ้าของงานวิจัยดั้งเดิมที่ได้อ้างอิง จำนวนของการอ้างอิงของงานวิจัยนั้นมีประโยชน์ในการหาความสำคัญของงานวิจัย เพราะงานวิจัยใดที่ถูกอ้างอิงเป็นจำนวนมากแสดงถึงคุณค่าของงานวิจัยนั้นๆได้เป็นอย่างดี

งานวิจัยเกี่ยวกับการสกัดเอกสารอ้างอิงในภาษาอังกฤษนั้นมีอยู่มากมายแต่ว่างานวิจัยในการสกัดเอกสารอ้างอิงภาษาไทยนั้นยังมีงานวิจัยอยู่อย่างจำกัด ซึ่งปัญหาของการสกัดเอกสารอ้างอิงภาษาไทยคือวิธีการเขียนไม่เหมือนกับในภาษาอังกฤษที่จะมีการแบ่งคำแต่ละคำอย่างชัดเจน ทำให้

การวิเคราะห์ส่วนประกอบในเอกสารอ้างอิงทำได้ยากกว่า แม้ว่าขั้นตอนวิธีในการสกัดเอกสาร จะมีการศึกษาที่หลากหลายแต่การสกัดเอกสารอ้างอิงสำหรับภาษาไทยแล้ว วิทยานิพนธ์ฉบับนี้จึงได้นำเสนอ การสกัดเอกสารอ้างอิงภาษาไทยด้วยไวยากรณ์ไม่พึ่งบริบท (Context Free Grammar) โดยเปรียบเหมือนเอกสารอ้างอิงเป็นภาษาๆหนึ่งที่ต้องมีไวยากรณ์ กำกับความถูกต้อง ซึ่งแต่ละส่วนประกอบในไวยากรณ์นั้น ก็คือชนิดของข้อมูลในเอกสารอ้างอิง

นอกจากการสกัดเอกสารอ้างอิงแล้ว ในส่วนของการทำดัชนีอ้างอิง (Citation Index) เพื่อหาความสำคัญของงานวิจัยนั้น ใช้วิธีเปรียบเทียบเอกสารว่าอ้างอิงถึงกันหรือไม่โดยใช้ชื่อผู้แต่ง ชื่อเรื่อง และปีที่พิมพ์ ในการทำดัชนี แต่จากการตรวจสอบบางครั้งพบว่าผู้ทำวิจัยเขียนชื่อเรื่องในการอ้างอิงไม่ถูกต้อง มีการพิมพ์คำตกหล่น หรือพิมพ์คำที่มีความหมายคล้ายคลึงกัน เช่น จริยธรรม จรรยาบรรณ วิทยานิพนธ์ฉบับนี้จึงนำเสนอเทคนิคของการหาค่าความเหมือนของสายอักขระ (Edit Distance) ในการเปรียบเทียบชื่อเรื่องระหว่างชื่อเอกสารที่ถูกอ้างอิงถึงกับชื่อเอกสารในฐานข้อมูล โดยใช้อัลกอริทึม Levenshtein distance ในการหาค่าความแตกต่างของชื่อเรื่องเมื่อมีค่าความแตกต่างไม่เกินค่าที่กำหนดไว้ แสดงว่าเอกสารนั้นเป็นเอกสารฉบับเดียวกัน

เป้าหมายของงานวิจัยนี้เป็นการสกัดเอกสารอ้างอิง โดยใช้ไวยากรณ์ไม่พึ่งบริบทในการสกัดส่วนประกอบของเอกสารอ้างอิงเพื่อใช้ในการทำดัชนีอ้างอิง (Citation Index) ของวิทยานิพนธ์ เพื่อนำผลที่ได้มาวิเคราะห์หาอัตราส่วนการอ้างอิงของงานวิจัย เช่น การอ้างอิงเอกสารภาษาไทยกับภาษาอังกฤษ ประเภทของเอกสารที่ใช้ในการอ้างอิงว่าผู้ทำวิจัยมีการค้นคว้าอย่างครอบคลุมหรือไม่ เพื่อหาคุณภาพของงานวิจัยต่อไป

งานวิจัยนี้ทำการสกัดเอกสารในขอบเขตของวิทยานิพนธ์ภาษาไทยโดยใช้อัลกอริทึมไวยากรณ์ไม่พึ่งบริบทเพราะเนื่องจากทราบโครงสร้างของการเขียนเอกสารอ้างอิงในวิทยานิพนธ์ที่เป็นมาตรฐานของบัณฑิตวิทยาลัยของมหาวิทยาลัยเกษตรศาสตร์จึงสามารถสร้างไวยากรณ์ขึ้นมาเพื่อใช้งานในการสกัดข้อมูลได้ทันที ต่างจากวิธีการสกัดด้วยการเรียนรู้จากข้อมูลที่ต้องทำการเตรียมข้อมูลเพื่อใช้ในการสอนระบบก่อน และเมื่อต้องการที่จะเพิ่มมาตรฐานใหม่ก็เพียงแต่เขียนไวยากรณ์เพิ่มเติม ทำให้สะดวกในการขยายไปใช้ในมาตรฐานอื่นๆ

ในงานวิจัยนี้ทำการสกัดเอกสารในขอบเขตของวิทยานิพนธ์ภาษาไทยของมหาวิทยาลัยเกษตรศาสตร์ในช่วงปี 2545-2549 และทำการรวบรวมผลของงานวิจัย เพื่อศึกษาประสิทธิภาพของอัลกอริทึมที่นำเสนอ เพื่อเป็นแนวทางในการพัฒนาต่อไป

วัตถุประสงค์

1. เพื่อสกัดเอกสารอ้างอิงภาษาไทยจากวิทยานิพนธ์ที่อยู่ในรูปแบบ พีดีเอฟ (PDF) โดยใช้วิธีการไวยากรณ์ไม่พึ่งบริบท

2. เพื่อศึกษาหาขั้นตอนวิธีในการประเมินประสิทธิผลของงานวิจัย เช่น จำนวนครั้งที่ถูกอ้างอิง ประเภทของเอกสารอ้างอิงในการเขียนวิทยานิพนธ์

ประโยชน์ที่คาดว่าจะได้รับ

1. ข้อมูลรายการเอกสารอ้างอิงที่ถูกสกัดและจัดเก็บในรูปแบบ XML เพื่อนำไปใช้ประมวลผลอื่นๆ ได้ทันที

2. สรุปการอ้างอิงของวิทยานิพนธ์ และนำไปวิเคราะห์ชนิดและปริมาณของการอ้างอิงในงานวิจัย เพื่อหาคุณลักษณะของการอ้างอิงวิทยานิพนธ์ในมหาวิทยาลัยเกษตรศาสตร์

ขอบเขตและข้อจำกัด

รูปแบบไฟล์ PDF ของงานวิจัยที่นำมาสกัดเอกสารอ้างอิงต้องเป็นไฟล์ PDF โดยในงานวิจัยนี้นำเฉพาะส่วนที่เป็นเอกสารอ้างอิงในวิทยานิพนธ์ โดยชนิดของ PDF ต้องเป็นแบบข้อความเท่านั้น ไม่สามารถสกัดในส่วนที่เป็นรูปภาพ เพราะในวิทยานิพนธ์ฉบับเก่าเป็นการถ่ายสำเนาจากรูปเล่มวิทยานิพนธ์แล้วทำการแปลงรูปภาพให้อยู่ในรูปแบบ PDF

รูปแบบของการเขียนเอกสารอ้างอิงที่จะทำการสกัดในงานวิจัยนี้นั้น เป็นวิทยานิพนธ์ของมหาวิทยาลัยเกษตรศาสตร์ ซึ่งใช้มาตรฐานของบัณฑิตวิทยาลัยจึงไม่รองรับการสกัดเอกสารอ้างอิงในรูปแบบอื่นๆที่นอกเหนือจากที่บัณฑิตวิทยาลัยกำหนดไว้ และทำการวิจัยเฉพาะเอกสารอ้างอิงที่เป็นภาษาไทยเท่านั้น

การเตรียมข้อมูลที่ใช้ในงานวิจัยนั้นผู้วิจัยทำการจัดทำข้อมูลชุดทดสอบ (Data Test) ขึ้นเอง เพราะว่าไม่มีชุดข้อมูลทดสอบเอกสารอ้างอิงภาษาไทยที่เป็นมาตรฐาน ต่างจากเอกสารอ้างอิงภาษาอังกฤษที่มีมาตรฐานกำหนดไว้ซึ่งเป็นข้อจำกัดของงานวิจัยชิ้นนี้

การเปรียบเทียบประสิทธิผลของอัลกอริทึม ทำการเปรียบเทียบกับ HMM (Hetzner, Erik. 2008) นั้นผลลัพธ์ที่แสดงออกมานั้นเป็นค่าที่วัดประสิทธิของโปรแกรมไม่ใช่ผลลัพธ์จากการสกัด ซึ่งนำมาเปรียบเทียบได้ในระดับหนึ่งเท่านั้น ซึ่งแนะนำให้ทำการเขียนโปรแกรมโดยใช้อัลกอริทึมดังกล่าวเองและนำผลที่ได้จากการสกัดมาเปรียบเทียบประสิทธิภาพ ส่วนโปรแกรม ParaCite (Mike Jewell. 2002) นั้นไม่สามารถที่จะนำมาใช้กับเอกสารภาษาไทยได้ จึงไม่ได้นำมาทดสอบ

การตรวจเอกสาร

ในส่วนนี้จะกล่าวถึงความรู้เบื้องต้นเกี่ยวกับการสกัดเอกสาร การแบ่งประเภทการสกัดเอกสารตามอัลกอริทึมที่ใช้และตัวอย่างงานวิจัย ความรู้เบื้องต้นเกี่ยวกับการทำดัชนีอ้างอิง และวิธีการเขียนเอกสารอ้างอิง

1. ความรู้เบื้องต้นเกี่ยวกับการสกัดข้อมูล

ในปัจจุบันข้อมูลถูกเก็บอยู่ในรูปแบบดิจิตอลอย่างแพร่หลายสามารถเข้าถึงได้ง่ายจากในอินเตอร์เน็ต ทำให้จำนวนเอกสารมีอัตราการเกิดขึ้นอย่างรวดเร็ว การสกัดข้อมูลจึงมีความจำเป็นเพื่อใช้ในการดึงข้อมูลเฉพาะส่วนที่ต้องการ เพื่อประหยัดเวลาในการทำงาน และลดขนาดข้อมูลในการจัดเก็บเอกสารทั้งฉบับก็จะเก็บข้อมูลเฉพาะที่สนใจ ซึ่งมีงานวิจัยในการสกัดข้อมูลอยู่หลายงานวิจัย ตัวอย่างงานที่เกี่ยวกับ การประมวลผลภาษาธรรมชาติ (NLP) เช่น การสกัดชื่อ (Name Entity Extraction) (Cohen and Sarawagi, 2004) หรือในงานสาขาอื่น เช่น Bioinformatics มีการสกัดชื่อโปรตีนในเอกสารงานวิจัยสาขา Biological (Fukuda *et al*., 1998) หรือจะเป็นงานทางด้าน Data Cleaning เช่น การสกัดข้อมูลจำเพาะที่จำเป็นในการทำงาน ซึ่งการสกัดเอกสารสามารถที่จะคัดกรองส่วนที่สำคัญ และขจัดส่วนที่ไม่ต้องการออกไป จากงานวิจัยที่กล่าวมามีวิธีการทำงานที่แตกต่างกันออกไปตามอัลกอริทึมที่ใช้ ซึ่งในการสกัดเอกสารมีรูปแบบข้อมูล และประเภทของการสกัดเอกสารดังนี้

1.1. รูปแบบของข้อมูล

ในปัจจุบันข้อมูลมีความสำคัญต่อระบบงานซึ่งข้อมูลนั้นจะสามารถนำมาใช้งานได้หรือไม่นั้นขึ้นอยู่กับรูปแบบของข้อมูลว่าถูกจัดเก็บไว้ในลักษณะใด ซึ่งข้อมูลที่จะนำไปใช้งานต้องผ่านการแปลงข้อมูลในรูปแบบที่เหมาะสม ก่อนนำไปใช้ประมวลผลหรือวิเคราะห์ด้วยคอมพิวเตอร์ ซึ่งรูปแบบของข้อมูลถูกแบ่งประเภทได้อยู่ 2 แบบ ได้แก่

1.1.1. ข้อมูลที่ไม่มีโครงสร้าง (Unstructured Data)

ข้อมูลที่ไม่มีโครงสร้างเป็นข้อมูลที่มีอยู่ทั่วไปในชีวิตประจำวัน เช่น บทความในหนังสือพิมพ์ เนื้อหาในเอกสารเวิร์ด และ เว็บไซต์ และอื่นๆซึ่งลักษณะของข้อมูลนั้นจะประกอบไปด้วย ข้อมูลที่เป็นสาระสำคัญกับข้อมูลที่เป็นส่วนเสริมที่จะอธิบายรายละเอียดต่างๆ ซึ่งการต้องการส่วนสำคัญของข้อมูลเฉพาะที่ผู้ใช้ให้ความสนใจนั้นก็คือการสกัดข้อมูล ซึ่งข้อมูลที่กล่าวมานั้นมีความสลับซับซ้อน ตัวอย่างเช่น หากต้องการทราบมีบุคคลใดบ้างที่ถูกกล่าวถึงในเอกสาร จำเป็นที่จะต้องทำการสกัดชื่อในเอกสารนั้นออกมา โดยข้อความที่เป็นชื่อนั้นจะไม่มีสิ่งใดบ่งบอกได้อย่างชัดเจนในเอกสาร เพราะว่าชื่อจะอยู่ปะปนในข้อความ ไม่มีตำแหน่งการจัดวางอย่างชัดเจน เพราะสามารถเป็นได้ทั้งประธานและกรรมในประโยค โดยชนิดของชื่อได้แก่ ชื่อคน,ชื่อบริษัท และชื่อสถานที่ ซึ่งทำให้การประมวลผลข้อความที่ไม่มีโครงสร้างนั้นทำได้ยาก เพราะไม่มีจุดที่บอกผู้ใช้ว่าเนื้อหาข้างในเอกสารส่วนใดเป็นส่วนที่ต้องการนำไปใช้งาน

ลักษณะของข้อมูลที่ไม่มีโครงสร้างในงานวิจัยที่เกี่ยวกับการสกัดข้อมูลนั้นคือข้อมูลที่ประกอบด้วย ข้อมูลที่ผู้ใช้ให้ความสนใจและข้อมูลที่ผู้ใช้ไม่ต้องการอยู่ปนกัน ไม่ได้ถูกแยกส่วนอย่างชัดเจน ซึ่งข้อมูลประเภทนี้จะสามารถนำไปประมวลผลได้ จำเป็นจะต้องผ่านการสกัดข้อมูลก่อน จึงมีงานวิจัยที่ทำ การสกัดข้อมูลชื่อ นามสกุล ที่อยู่ และเบอร์โทรศัพท์ จากเนื้อหาอีเมลและเว็บไซต์ (Culotta *et al*., 2004) เมื่อผ่านการสกัดข้อมูล ข้อมูลในส่วนนี้จะมีชนิดข้อมูลมากำกับด้วย ว่าภายในเนื้อหานั้นข้อมูลแต่ละส่วนมีชนิดข้อมูลเป็นอะไร เพื่อสามารถนำไปประมวลผลต่อได้

ตัวอย่างข้อมูลที่ไม่มีโครงสร้าง

นายวีรพันธ์ พูลเกษ กรรมการผู้จัดการ บริษัท ไทคอน อินดัสเทรียล คอนเน็คชั่น จำกัด (มหาชน) หรือ TICON กล่าวว่า บริษัทฯ ตั้งเป้ารายได้รวมและกำไรในปี 2553 เติบโต 20% จากปีก่อน โดยในปีก่อนรายได้ 2,529.20 ล้านบาท และกำไรสุทธิ653.28 ล้านบาท โดยรายได้จากค่าเช่าพื้นที่ปัจจุบันยังไม่ได้รับผลกระทบมากนัก จากสถานการณ์ทางการเมืองภายในประเทศ โดยพื้นที่ส่วนใหญ่อยู่แถว จ.ระยอง และ จ.ชลบุรี ตราบใดที่ยังสามารถส่งออกและนำเข้าได้ตามปกติก็คงไม่มีปัญหา.

ที่มา : http://www.stockwave.in.th/top-headline/10781-news-140510.html

จากตัวอย่างข้อความจะมีชื่อคน และ ชื่อสถานที่ ปนอยู่ในข้อความ โดยข้อมูลที่ไม่มีโครงสร้างนั้นจะไม่มีการระบุชนิดข้อมูล ทำให้ไม่สามารถนำมาประมวลผลได้ทันทีต้องทำการสกัดข้อมูลออกมาก่อนดังตัวอย่าง

ตัวอย่างข้อมูลที่ไม่มีโครงสร้างที่ถูกสกัดข้อมูลแล้ว

นาย<Tag TYPE="Name">วีรพันธ์ พูลเกษ</Tag> กรรมการผู้จัดการ บริษัท <Tag TYPE="Name">ไทคอน อินดัสเทรียล คอนเน็คชั่น</Tag> จำกัด (มหาชน) หรือ <Tag TYPE="Name">TICON</Tag> กล่าวว่า บริษัทฯ ตั้งเป้ารายได้รวมและกำไรในปี 2553 เติบโต 20% จากปีก่อน โดยในปีก่อนรายได้ 2,529.20 ล้านบาท และกำไรสุทธิ653.28 ล้านบาท โดยรายได้จากค่าเช่าพื้นที่ปัจจุบันยังไม่ได้รับผลกระทบมากนัก จากสถานการณ์ทางการเมืองภายในประเทศ โดยพื้นที่ส่วนใหญ่อยู่แถว จ.<Tag TYPE="Name">ระยอง</Tag> และ จ.<Tag TYPE="Name">ชลบุรี</Tag> ตราบใดที่ยังสามารถส่งออกและนำเข้าได้ตามปกติก็คงไม่มีปัญหา

1.1.2. ข้อมูลที่มีโครงสร้าง (Structured Data)

ข้อมูลที่มีโครงสร้างเป็นข้อมูลที่มีความสำคัญมากในการทำงานด้านวิศวกรรมซอฟต์แวร์ เนื่องจากเป็นข้อมูลที่มีระเบียบและมีความถูกต้องของข้อมูล สามารถนำไปใช้ประมวลผลด้วยคอมพิวเตอร์ได้ทันที เพราะข้อมูลที่ถูกเก็บนั้นจะสามารถระบุชนิดข้อมูลได้ เช่น ตัวเลข (Integer), เลขทศนิยม (Float), ข้อความ (String), วันที่ (Date) และอื่นๆ ซึ่งชนิดข้อมูลดังกล่าวสามารถกำหนดขนาดของข้อมูลได้ เช่น ตัวเลขต้องมีค่าระหว่าง 1 ถึง 500, ข้อความมีขนาดไม่เกิน 24 ตัวอักษร ซึ่งที่กล่าวมานั้นเป็นชนิดของข้อมูล เมื่อกำหนดชนิดของข้อมูลก็ทำการตั้งชื่อของชนิดข้อมูลที่กำหนด เช่น ในใบสั่งซื้อจะประกอบด้วยประเภทข้อมูล รหัสสินค้า ชื่อสินค้า จำนวน ส่วนลด ซึ่งประเภทข้อมูลดังกล่าวก็จะมีชนิดของข้อมูลแตกต่างกันไป

ลักษณะพิเศษของข้อมูลที่มีโครงสร้างเป็นข้อมูลที่ในแต่ละส่วนนั้นถูกระบุชนิดข้อมูล (Metadata) อย่างชัดเจน เช่น ข้อมูลในฐานข้อมูล ได้แก่ MS Access, SQL Server, Foxpro, Oracle, My SQL และไฟล์ XML ซึ่งข้อมูลจะมีความถูกต้องเพราะในแต่ละส่วนจะมีการ

กำหนดชนิดข้อมูลส่วนต่างๆไว้แล้ว เช่น XML ไฟล์จะมีส่วนที่เป็น DTD ที่ใช้ในการตรวจสอบว่าชนิดของข้อมูลที่เก็บนั้นถูกต้องหรือไม่ ดังตัวอย่างต่อไปนี้

ตัวอย่างไฟล์ XML

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE people_list SYSTEM "example.dtd">
<paper_list>
    <paper>
      <authors>
    <author>Giles, C. </author>
    <author>Lee and Bollacker </author>
    <author> Lawrence, Steve </author>
<authors>
<title>CiteSeer: an automatic citation indexing system</title>
<year>1998</year>
<page>89-98</page>
<publisher>ACM</publisher>
<address>New York, NY, USA</address>
    </paper>
</paper_list>
```

ตัวอย่างไฟล์ DTD

```
<!ELEMENT paper_list (paper*)>
<!ELEMENT paper (authors, title, year?,page?,publisher?,address?)>
<!ELEMENT authors (author+)>
<!ELEMENT title (#PCDATA)>
<!ELEMENT year (#PCDATA)>
<!ELEMENT page (#PCDATA)>
<!ELEMENT publisher (#PCDATA)>
<!ELEMENT address (#PCDATA)>
<!ELEMENT author (#PCDATA)>
```

ข้อดีของข้อมูลที่มีโครงสร้างนั้นสามารถนำไปใช้งานได้ง่าย โดยเชื่อมต่อกับผู้ใช้ผ่าน โปรแกรมประเภทจัดการฐานข้อมูล (DBMS: Database Management System) ที่เป็นตัวกลางในการติดต่อ ซึ่งประโยชน์ของการทำงานกับข้อมูลที่มีโครงสร้าง คือ ลดข้อมูลที่ซ้ำซ้อน, ข้อมูลที่ถูกเก็บมีความถูกต้อง สามารถนำข้อมูลไปขยายการทำงานได้ง่ายและหลากหลายรูปแบบ

1.1.3. ข้อมูลกึ่งโครงสร้าง (Semi-Structured Data)

ข้อมูลกึ่งโครงสร้างเป็นข้อมูลที่มีการผสมกันระหว่าง ข้อมูลที่มีโครงสร้างและข้อมูลที่ไม่มีโครงสร้าง โดยในข้อมูลนั้นจะมีโครงสร้างแต่มีบางตำแหน่งของข้อมูลในโครงสร้างที่เป็นสายอักขระยาว หากต้องการที่จะใช้ข้อมูลประเภทนี้ในการประมวลผลต้องทำการสกัดข้อมูลออกมาก่อน ข้อมูลกึ่งโครงสร้างนั้นก็ได้แก่ อีเมล, สายอักขระ DNA และอื่นๆ ตัวอย่างเช่น ข้อมูลของอีเมลนั้นจะมีส่วนประกอบของข้อมูลที่มีโครงสร้างได้แก่ ข้อมูลผู้ส่ง ผู้รับ วันที่ และชื่อเรื่องของอีเมลฉบับนั้นๆ ข้อมูลที่กล่าวมานั้นจะถูกเก็บอยู่ในรูปแบบ ส่วนข้อมูลที่ไม่มีโครงสร้างนั้นก็จะเป็นเนื้อหาของอีเมลที่ไม่มีข้อกำหนดอย่างชัดเจน

ตัวอย่าง ข้อมูลอีเมลประกอบไปด้วย ข้อมูลส่วนหัว (Header Message) ที่ประกอบไปด้วยผู้ส่ง ผู้รับ และเส้นทางของอีเมล ซึ่งข้อมูลบางอย่างนั้นผู้ใช้งานไม่ได้แสดงให้ผู้ใช้งานเห็น แต่จะเป็นส่วนที่โปรแกรมใช้ในการทำงานซึ่งข้อมูลดังกล่าวนั้นสามารถนำไปประมวลผลด้วยคอมพิวเตอร์ได้ทันที

From owner-is-all-compcont@bham.ac.uk Fri Aug 18 15:10:01 2000

Received: from bham.ac.uk by isdux1.bham.ac.uk (8.8.8/1.1.8.2/14Aug95-0452PM)

id PAA0000016479; Fri, 18 Aug 2000 15:10:00 +0100 (BST)

Received: from isdugp.bham.ac.uk ([147.188.128.15] helo=isdux1.bham.ac.uk)

by bham.ac.uk with esmtp (Exim 3.16 #3)

id 13PmpM-0000XA-00; Fri, 18 Aug 2000 15:08:56 +0100

Received: by isdux1.bham.ac.uk (8.8.8/1.1.8.2/14Aug95-0452PM)

id PAA0000009231; Fri, 18 Aug 2000 15:09:56 +0100 (BST)

Message-Id: <200008181409.PAA0000009231@isdux1.bham.ac.uk>

Subject: Netscape vulnerability fix

To: all-compcont@bham.ac.uk

Date: Fri, 18 Aug 2000 15:09:56 +0100 (BST)

From: Chris Bayliss <C.B.Bayliss@bham.ac.uk>

Reply-To: C.B.Bayliss@bham.ac.uk

X-Mailer: ELM [version 2.4 PL25]

MIME-Version: 1.0

Content-Type: text/plain; charset=US-ASCII

Content-Transfer-Encoding: 7bit

Sender: owner-is-all-compcont@bham.ac.uk

Precedence: bulk

Status: RO

Netscape have released a new version of Netscape Communicator (4.75) which is not subject to the Java vulnerability announced preciously on this list.

If you use a vulnerable version of Netscape (version 4.0 - 4.74) it is

recommended that you upgrade.

Chris Bayliss

Information Services

ที่มา : http://www.email.bham.ac.uk/intro_str.shtml

1.2. การสกัดข้อมูล

การสกัดข้อมูล (Data Extraction) จุดมุ่งหมายก็เพื่อคัดแยกข้อมูลที่ต้องการออกจากข้อมูลไม่มีรูปแบบ (Unstructured Data) และจัดเก็บให้มีรูปแบบ(Structured Data) เพราะว่าในปัจจุบันจำนวนของเอกสารที่ไม่รูปแบบนั้นเติบโตเป็นอย่างมาก การสกัดข้อมูลจึงจำเป็นเพื่อที่จะทำให้เอกสารที่ไม่มีรูปแบบนั้นถูกจัดเก็บสามารถนำไปใช้ประมวลผลได้ทันที ซึ่งในอดีตเอกสารมีจำนวนไม่มากการใช้แรงงานคนในการอ่านและจัดเก็บข้อมูลนั้นสามารถทำได้ แต่เมื่อมีเอกสารเป็นจำนวนมากขึ้นการสกัดเอกสารด้วยคอมพิวเตอร์จึงเป็นสิ่งจำเป็นเพราะมีการทำงานที่ความสะดวกและรวดเร็วกว่า

ระบบที่จำเป็นต้องใช้งานการสกัดเอกสารนั้นส่วนมากเป็นระบบที่ต้องการทำงานแบบอัตโนมัติกับข้อมูลที่มีเป็นจำนวนมากซึ่งหากใช้แรงงานมนุษย์นั้นจะใช้เวลานานเช่น ระบบการสกัดข้อมูลจากเว็บไซต์ เว็บไซต์มีจำนวนมากหากใช้แรงงานมนุษย์ในการทำงานเปิดเว็บเพจอ่าน และสกัดข้อมูลจะต้องใช้เวลาเป็นจำนวนมาก

การสกัดเอกสารอ้างอิงนั้นมีวิธีการทำงานหลายรูปแบบแต่จะมีวิธีการหลักๆ ในการทำงานขั้นตอนที่เหมือนกัน คือจะแบ่งเอกสารเป็นส่วนๆที่เรียกว่า Token โดยมีวิธีการแบ่งเอกสารนั้นจะมีส่วนประกอบอยู่ 2 ส่วนได้แก่การทำนิพจน์ปรกติ (Regular Expression) และ Dictionary Search ดังนี้

1.2.1 นิพจน์ปรกติ (Regular Expression)

นิพจน์ปรกติ คือ นิพจน์ที่ใช้ในการเขียนอธิบายสายอักขระเฉพาะเรื่อง เพื่อใช้ในการค้นหาข้อความที่ต้องการ ซึ่งในงานสกัดเอกสารนั้น Regular Expression ช่วยในการระบุว่าข้อความบางประเภทที่ต้องการ เช่น นิพจน์ของอีเมลและที่อยู่เว็บไซต์

Email = /^([a-zA-Z0-9_\.\-])+\@(([a-zA-Z0-9\-])+\.)+([a-zA-Z0-9]{2,4})+$/

URL = (([a-zA-Z0-9\-])+\.)+([a-zA-Z0-9]{2,4})+$/

เมื่อใช้นิพจน์เทียบกับข้อความที่ต้องการเปรียบเทียบแล้วพบว่า match กันก็จะสามารถระบุชนิดข้อมูลนั้นตามประเภทของ pattern ซึ่งการทำ Regular Expression นั้นจะสามารถ

แบ่งข้อมูลที่เป็นสายอักขระยาวๆออกเป็นส่วนย่อยตามรูปแบบนิพจน์ที่ได้กำหนดไว้ ในส่วนของชนิดข้อมูลนั้น ก็จะเป็นชนิดข้อมูลกว้างๆยังไม่สามารถเฉพาะเจาะจงได้ว่าข้อมูลเป็นประเภทอะไร

1.2.2 Dictionary Search

การใช้ Dictionary Search นั้นเป็นการเทียบข้อมูลใน Dictionary ที่ได้กำหนดไว้กับข้อความที่รับเข้ามาซึ่งจะแตกต่างกับการนิพจน์ปรกติที่มีความยืดหยุ่นมากกว่าแต่ Dictionary search นั้นเหมาะกับการใช้เปรียบเทียบในรูปแบบ keyword หรือคำสำคัญเพื่อหาชนิดข้อมูล ตัวอย่าง เช่น

Month = {“มกราคม”,”กุมภาพันธ์”, ... , “ธันวาคม”}
City = {“กรุงเทพ”,”ขอนแก่น”, ... , “อ่างทอง”}

เมื่อเทียบข้อมูลแล้วตรงกับค่าใน Dictionary ก็สามารถระบุได้ว่าข้อมูลนั้นเป็นชนิดข้อมูลใด ซึ่งถูกนำมาใช้สกัดข้อมูลทางด้านการประมวลผลภาษาธรรมชาติ (Riloff and Lehnert, 1993) เมื่อจบการทำงานในส่วนของ Dictionary Search ระบบจะมีข้อมูลสั้นๆที่ถูกระบุประเภทของข้อมูล

การแบ่งข้อมูลเป็นส่วนๆ (Token) ดังวิธีการที่ผ่านมา ก็จะเข้าสู่ขั้นตอนของการสกัดข้อมูลซึ่งจะมีวิธีการทำงานอยู่หลายรูปแบบตามอัลกอริทึมในการทำงาน เมื่อวิเคราะห์การทำงานของแต่ละงานวิจัย สามารถที่จะจัดกลุ่มการสกัดข้อมูลตามวิธีการทำงานได้ 3 รูปแบบได้แก่ การสกัดเอกสารแบบเรียนรู้ข้อมูล (Learning based), การสกัดเอกสารแบบเปรียบเทียบรูปแบบ (Template based) และการสกัดเอกสารแบบการใช้กฎ (Rule based)

1.2.1. การสกัดเอกสารแบบเรียนรู้ข้อมูล (Learning Based)

การสกัดเอกสารแบบเรียนรู้ข้อมูลเป็นวิธีการของ machine learning ซึ่งใช้หลักการเรียนรู้จาก ข้อมูลสอนระบบ (Training Data) ซึ่งในงานประเภทสกัดข้อมูล (Extract Information) ใช้วิธีที่เรียกว่า statistical machine learning ที่ให้ระบบเรียนรู้ข้อมูลที่มีผลลัพธ์ เพื่อเก็บค่าสถิติต่างๆของข้อมูล เพื่อใช้ในการทำงานกับเอกสารที่ยังไม่ถูกสกัด ในการเรียนรู้ระบบระบบต้องการข้อมูลที่ใส่รายการอ้างอิงที่มีผลลัพธ์กำกับโดยมีการระบุชนิดข้อมูลในแต่ละส่วน

(Training Data) เพื่อให้อัลกอริทึมนำข้อมูลมาวิเคราะห์หาค่าสถิติในลำดับการเปลี่ยนแปลงของชนิดข้อมูลแต่ละส่วนรายการอ้างอิง ซึ่งการใช้วิธี Learning Based นั้นมีอัลกอริทึมในการทำงานอยู่หลายอัลกอริทึมซึ่งอาจจะเป็นการปรับปรุงอัลกอริทึมอื่นหรือใช้วิธีการประยุกต์วิธีการต่างๆ เข้ามาใช้เพื่อประสิทธิภาพของระบบ ซึ่งใน machine learning นั้นมีอัลกอริทึมที่ใช้ดังต่อไปนี้

ก. Markov Model

Markov Model (Alvin, 1967) เป็นโมเดลที่อธิบายถึงความน่าจะเป็นที่จะเปลี่ยนสถานะหนึ่งไปอีกสถานะหนึ่ง โดยในการสกัดข้อมูล สถานะก็จะเป็นเหมือนชนิดข้อมูลที่ต้องการสกัด ส่วนความน่าจะเป็นนั้นเกิดจากการนำข้อมูลที่มีผลลัพธ์ (Training Data) มาเข้าระบบการเรียนรู้เพื่อหาค่าสถิติของความน่าจะเป็นในการเรียงลำดับของชนิดข้อมูลเข้าว่ามีการเรียงลำดับอย่างไร

โดยที่ N คือจำนวนสถานะ ตั้งแต่ s_1 ถึง s_n โดยที่ s_i เป็นสถานะต้นทาง และ s_j เป็นสถานะปลายทาง ซึ่งในงานวิจัยนี้ S เป็นเสมือนส่วนประกอบของเอกสารอ้างอิง เช่น ชื่อผู้แต่ง ชื่อเรื่อง ปีที่พิมพ์ สถานที่พิมพ์ ชื่อวารสาร เล่มที่ และหน้า โดยมีลำดับของข้อมูลเข้าเป็น $q_1, q_2, \dots, q_t$ ได้แก่ เอกสารอ้างอิงที่ต้องการสกัดโดยจะแบ่งเอกสารออกเป็นส่วนๆ ตามสัญลักษณ์ที่กำหนดไว้ ก็จะได้สมการคือ

$$P(q_t = S_j | q_{t-1} = S_i, q_{t-2} = S_k, \dots)$$

Markov จึงได้สรุปสมการในรูปแบบย่อได้ดังนี้

$$P(q_t = S_j | q_{t-1} = S_i, q_{t-2} = S_k, \dots) \approx P(q_t = S_j | q_{t-1} = S_i)$$

ซึ่งสามารถเขียนเป็นสมการของความน่าจะเป็นของลำดับข้อมูล $S_1, \dots, S_n$ ได้ดังนี้

$$P(S_1, \dots, S_n) = \prod_{t=1}^{n} P(S_t | S_{t-1})$$

โดยเริ่มต้นจะมีเมทริกซ์ A เป็นเมทริกซ์ค่าความน่าจะเป็น มีค่า a_{ij} ที่ถูกกำหนดไว้เป็นค่าความน่าจะเป็นในการเปลี่ยนสถานะจาก i ไปเป็น สถานะ j ซึ่งหมายถึง ความน่าจะเป็นที่ข้อมูลจะเปลี่ยนสถานะจากชนิดข้อมูลหนึ่งไปเป็นอีกชนิดข้อมูลหนึ่ง

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1j} & \cdots & a_{1N} \\ a_{21} & a_{22} & \cdots & a_{2J} & \cdots & a_{2N} \\ \vdots & \vdots & \cdots & \vdots & \cdots & \vdots \\ a_{j1} & a_{i2} & \cdots & a_{ij} & \cdots & a_{iN} \\ \vdots & \vdots & \cdots & \vdots & \cdots & \vdots \\ a_{N1} & a_{N2} & \cdots & a_{Nj} & \cdots & a_{NN} \end{bmatrix}$$

โดยที่

$$a_{ij} = P(\, q_t = j \mid q_{t-1} = i) \; ; 1 \leq i, j \leq N$$

$$a_{ij} \geq 0 \; ; \forall i, j$$

$$\sum_{j=1}^{N} a_{ij} = 1 \; ; \forall i$$

โดยที่ $\boldsymbol{\pi}_3$ นั้นมีค่า

$$\pi_i = P[q_1 = S_i]; \; 1 \leq i \leq N$$

ข. Hidden Markov Model (HMM)

Hidden Markov Model (Rabiner, L.R, 1989) เป็นโมเดลทางสถิติที่พัฒนามาจาก Markov Model โดยทำการเพิ่มสถานะที่ไม่สามารถมองเห็นได้ (unobserved state) ซึ่ง Markov Model นั้นจะทราบการเปลี่ยนสถานะโดยตรงจึงมีค่าความน่าจะเป็นเพียงค่าเดียวที่จะบอกว่าสถานะถัดไปนั้นควรจะเป็นอะไร แต่ใน Hidden Markov Model นั้นจะมีสถานะอีกหนึ่งสถานะที่ช่วยในการกระจายความน่าจะเป็นของการเปลี่ยนสถานะที่น่าจะเกิดขึ้นอีกหนึ่งค่า

การออกแบบระบบประกอบด้วยส่วนต่างๆ ซึ่งมีค่าดังนี้

N = จำนวน hidden states ใน Model

M = จำนวน observation symbols ใน Model

S = {1,2,…,N} เซ็ตของ hidden states

V = {1,2,…,M} เซ็ตของ observation symbols

A = $\{a_{ij}\}$ เป็นเมทริกซ์ความน่าจะเป็นในการเปลี่ยนสถานะ

โดยที่จากเดิมสมการของ Markov

$$P(S_1, \ldots, S_n) = \prod_{t=1}^{n} P(S_t|S_{t-1})$$

เมื่อเพิ่มตัวแปรในการทำงานใน Hidden Markov Model ที่เป็นค่า V นั้น จะได้ค่า $P(S_1, \ldots, S_n | V_1, \ldots, V_m)$ สามารถที่จะใช้กฎของ Baye's ได้ดังนี้

$$P(S_1, \ldots, S_n | V_1, \ldots, V_m) = \frac{P(S_1, \ldots, S_n | V_1, \ldots, V_m) P(S_1, \ldots, S_n)}{P(V_1, \ldots, V_m)}$$

โดยที่

$$a_{ij} = P(q_{t+1} = S_j \mid q_t = S_i), \qquad 1 \leq i, j \leq N$$

B = ความน่าจะเป็นของ observation symbols ในสถานะ ,

$B = \{ b_j(k) \}$

$$b_j(k) = P(v_k \; at \; t \mid q_t = S_j), \qquad 1 \leq j \leq N, \; 1 \leq k \leq M$$

สถานะเริ่มต้น

$$\pi_i = P(q_1 = S_i)$$

สรุปโมเดลประกอบด้วย

$$\lambda = (A, B, \pi)$$

HMM ถูกใช้ในการสกัดเอกสารงานวิจัยทั้งฉบับ ซึ่งประกอบด้วยชนิดของข้อมูลปัจจุบัน เช่น ชื่อเรื่อง, ชื่อผู้แต่ง, อีเมล, วันที่, ที่อยู่, บทคัดย่อ และ คำสำคัญ เป็นสถานะ (Seymore *et al*. 1999) นอกจากนี้ยังมีงานวิจัยที่ได้มีการนำ HMM มาใช้สกัดเฉพาะรายการเอกสารอ้างอิง โดยในการเรียนรู้ระบบใช้รายการอ้างอิงของ Cora Dataset ที่ถูกแท็กชนิดข้อมูลแล้วทั้งหมด 350 เอกสารอ้างอิง และทดสอบระบบด้วยข้อมูล 142 เอกสารอ้างอิง (Erik, 2008) ซึ่งข้อมูลที่นำมาประมวลผลนั้นใช้เอกสารที่เป็นรูปแบบ LaTaX ไฟล์

ค. Support Vector Machines (SVM)

SVM เป็นอัลกอริทึมที่ใช้หลักการของสถิติทำการจัดหมวดหมู่ข้อมูลที่ต้องการ โดยอาศัย ข้อมูลสอนระบบ ซึ่งถูกนำไปประยุกต์ใช้ในงานหลายสาขา เช่น มีการสกัดชื่อยาในเอกสารทางการแพทย์ (Koichi and Collier, 2002) การแบ่งประเภทของ Texture (Texture classification) (Li *et al*. 2003) และการสกัดเอกสารอ้างอิง (Hui *et al*. 2003) ซึ่งในงานวิจัยการสกัดเอกสาร โดยใช้ SVM มีหลักการทำงาน คือ ข้อความที่ถูกแบ่งเป็นช่วงจะถูกกำหนดมิติลงในกราฟ โดยที่ข้อความนั้นมีอยู่ในมิติของคลาสกลุ่มใดก็จะถูกกำหนดเป็นชนิดข้อมูลนั้น ซึ่งการสอนระบบประกอบด้วยคู่ของข้อมูลเรียนรู้ ที่เป็น Input ซึ่งได้แก่เอกสารอ้างอิงที่ยังไม่ถูกสกัด และ Output ของระบบที่เป็นเอกสารอ้างอิงที่ถูกทำการระบุชนิดของส่วนประกอบในเอกสารอ้างอิงแล้ว

$$\{(x_1, y_1), \ldots, (x_n, y_n)\}$$

เพื่อนำคู่ของข้อมูลไปสร้างฟังก์ชั่นของการแบ่งประเภท ซึ่งฟังก์ชั่น $f : X \rightarrow R$ เป็นฟังก์ชั่นในการเพื่อหาตำแหน่ง N ในกราฟ โดยที่อินพุต $x \in X$ ซึ่งได้แก่เอกสารอ้างอิงที่ถูกแบ่งออกเป็นส่วนๆ จะถูกกำหนด label $y \in Y$ ที่ค่า $Y = \{+1, -1\}$ ซึ่งได้แก่ชนิดของข้อมูลในเอกสารอ้างอิง เช่น

$$(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n) \subset R^N \times \{\pm 1\}$$

ง. Conditional Random Fields (CRF)

CRF เป็น probabilistic framework ที่ใช้สำหรับแบ่งและระบุชนิดข้อมูล โดยอาศัยเงื่อนไขที่มีลักษณะโครงสร้างแบบกราฟที่ไม่มีทิศทาง โดยมีเงื่อนไขเป็นค่า x และค่า

random variable ที่เป็นตัวแทนของ observation sequeces โดยกำหนดให้กราฟ $G = (V, E)$ เป็นกราฟไม่มีทิศทางมีโหนด $v \in V$ ที่เป็นค่าของแต่ละ random variables แทนด้วย Y_v ของ Y โดยแต่ละ Y_v มีคุณสมบัติเดียวกันกับ Markov Model ในกราฟ G จะได้ (X, Y)

โดยที่ $G = (V, E)$ เป็นกราฟที่มีค่า $Y = (Y_v)_{v \in V}$ โดยที่ Y เป็นโหนดของกราฟ G แล้ว (X, Y) เป็น Condition random field เมื่อเงื่อนไขคือ X และมีค่าสุ่มที่ Y_v และใช้คุณสมบัติของ Markov เพื่อสร้างกราฟดังนี้

$$p(Y_v|X, Y_w, w \neq v) = p(Y_v|X, Y_w, w \sim v)$$

โดยที่ $w \sim v$ นั้นหมายถึง โหนด w และโหนด v นั้นอยู่ใกล้กันในกราฟ G

เมื่อกำหนด linear-chain CRF พารามิเตอร์ $\lambda = \{\gamma, \dots\}$ ให้กับความน่าจะเป็นของสถานะ $Y = y_1 \dots y_T$ ที่รับอินพุต $X = x_1 \dots x_T$ จะได้

$$P_\gamma(Y|X) = \frac{1}{Z_x} \exp\left(\sum_{t=1}^{T} \sum_{k} \gamma_k f_k(y_{t-1}, y_t, X, t)\right)$$

ประโยชน์ของ CRF ที่ดีกว่า HMM คือ เงื่อนไขเป็นแบบปรับเปลี่ยนได้ ทำให้ผลลัพธ์ที่ได้นั้นไม่ขึ้นอยู่กับค่าที่กำหนดคงที่ในแบบ HMM และ MEMM ดังรูปที่ 3 อีกทั้ง CRF นั้นไม่มีปัญหาเรื่องไบแอส ด้วยเหตุนี้ CRF นั้นจึงถูกนำไปใช้ในการทำ Sequence labeling มากกว่า MEMM และ HMM

CRF ถูกนำมาใช้พัฒนาโปรแกรมการสกัดเอกสารอ้างอิง เช่น พัฒนาระบบ ParsCit (Isaac G. Councill *et al.* 2008) ที่เป็นการสกัดเอกสารอ้างอิงโดยใช้วิธีการ Conditional Random Field (CRF) โดยใช้ข้อมูลของ Cora Dataset ซึ่ง ParsCit ในรุ่นแรกนั้นใช้ Maximum Entropy (ME) ในการสกัดเอกสาร แต่เปลี่ยนมาใช้ CRF แทน โดยปัจจุบันการทำงานของ ParsCit ถูกนำไปใช้ในระบบ CiteseerX จากนั้น C. Shoemaker (2009) พัฒนาระบบ FreeCite ที่เป็น Citation Parser โดยใช้วิธีการ CRF เหมือนกับระบบ ParsCit แต่นำมาปรับปรุงโดยภาษา Ruby และยังมีการใช้ CRF ในการสกัดข้อมูลในงานวิจัยอื่นๆ อาทิเช่น การสกัดข้อมูลจากเอกสาร

งานวิจัย (Fuchun and McCallum, 2006) การสกัดตารางจาก HTML (David *et al*. 2003) และการสกัด Named Entity (McCallum and Li, 2003)

1.2.2. การสกัดเอกสารแบบเปรียบเทียบรูปแบบ (Template Based)

เป็นการสกัดเอกสารโดยใช้หลักการเปรียบเทียบข้อมูลกับรูปแบบที่กำหนดไว้ ซึ่งรูปแบบในการเขียนเอกสารอ้างอิงนั้นจะมีอยู่หลายรูปแบบตามมาตรฐานต่างๆ ที่กำหนดไว้ ซึ่งเราสามารถใช้รูปแบบของแต่ละมาตรฐานมาเป็น Template ในการสกัดเอกสารอ้างอิง อาทิเช่น ระบบ ParaCite (Jewell, 2002) ระบบ BibPro (Chien-Chih *et al*. 2008) และ การทำ Template Mining (Ying *et al*. 1999)

โดยมีโครงสร้างการทำงานหลักอยู่ 2 ส่วนดังต่อไปนี้

1. ส่วนสร้าง Template

ทำหน้าที่ในการสร้างรูปแบบเพื่อใช้ในการเปรียบเทียบ โดยระบบจะนำเอกสารอ้างอิงที่มี Metadata กำกับอยู่แล้ว เช่น BibTex นำมาสร้างเป็น Template โดยเก็บรวบรวมจาก Google, IEEE, ACM และ CiteSeer ซึ่ง Template จะถูกเก็บไว้ในฐานข้อมูล Template เพื่อใช้ในการเปรียบเทียบและสกัดเอกสารในส่วนถัดไป เช่น ระบบ ParaCite (Jewell, 2002) จะเก็บรูปแบบของ Template ดังต่อไปนี้

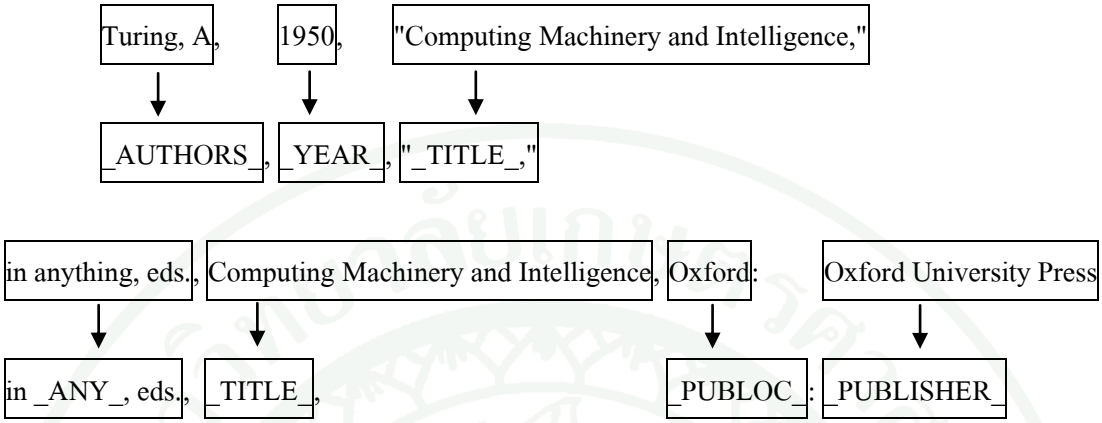
เอกสารอ้างอิง

Turing, A, 1950, "Computing Machinery and Intelligence," in anything, eds., Computing Machinery and Intelligence, Oxford: Oxford University Press

Template

AUTHORS, _YEAR_, "_TITLE_," in _ANY_, eds., _TITLE_, _PUBLOC_: _PUBLISHER_

ซึ่งสามารถที่จะจับคู่สายอักขระในเอกสารอ้างอิงกับประเภทข้อมูลใน Template ได้ดังนี้



2. ส่วนสกัดเอกสาร

ในการสกัดเอกสาร ระบบจะนำรายการเอกสารอ้างอิงมา Parsing และจับคู่ข้อมูลที่แบ่งกับสัญลักษณ์ เพื่อนำไปหาความเหมือนกับ Template ในฐานข้อมูลว่ามีความคล้ายกับ Template ใดมากกว่ากัน และนำ Template มาเป็นรูปแบบในการสกัดเอกสารว่าในรายการอ้างอิงมีส่วนประกอบอะไรบ้าง และมีการจัดเรียงลำดับอย่างไร อาทิเช่น M. Jewell (2004) ได้พัฒนาระบบ ParaCite โดยใช้วิธีการของ Regular Expression ร่วมกับ Template ในการทำงาน ซึ่งระบบมี Template ทั้งหมด 385 รูปแบบโดยใช้โมดูลของโปรแกรมภาษา Perl ชื่อ Biblio Citation Parser ในการสกัดส่วนต่างๆของเอกสารอ้างอิง ระบบสามารถรองรับการเขียนนามสกุลผู้แต่งที่มีหลายคำ (Multiple-word surname) และ ชื่อและนามสกุลผู้แต่งที่เขียนย่อด้วยอักษรตัวเดียว จากการทำงานนั้นพบว่าระบบไม่สามารถที่จะสกัดชื่อผู้แต่งของเอกสารที่มีผู้แต่งหลายคนได้ ซึ่งอาจจะทำให้การสกัดส่วนอื่นๆ ผิดพลาดด้วย ในส่วนของการจับคู่นั้นก็จะมีอัลกอริทึมในการเปรียบเทียบ เช่นในระบบ BibPro (Chien-Chih *et al*. 2008) นำเสนอเทคนิค Sequence Alignment ในการจับคู่รายการเอกสารอ้างอิงกับ Template ส่วน ParaCite นั้นจะเป็นระบบให้น้ำหนักของแต่ละส่วนในเอกสารและเลือก Template ที่มีน้ำหนักรวมดีที่สุด

1.2.3. การสกัดเอกสารแบบการใช้กฎ (Rule Based)

ส่วนของ Rule Based จะเป็นการทำงานด้วยกฎที่กำหนดไว้ตอนเริ่มใช้งานไม่มีการเปลี่ยนแปลงตามข้อมูลที่เข้ามา ซึ่ง Rule Based นั้นนิยมใช้งานเพราะว่าง่ายต่อการพัฒนาและไม่ต้องการ Training Data ซึ่งการทำงานของ Rule Based นั้นสามารถนำไปใช้ร่วมกับวิธีการอื่นเพื่อเพิ่มประสิทธิภาพการทำงานของระบบ ซึ่งวิธีการทำงานแบบ Rule Based นั้นก็มีวิธีการแบบ Heuristic Rule

ในการสกัดข้อมูลนั้น Heuristic Rule นั้นเป็นเทคนิคการสร้างกฎจากประสบการณ์ที่ได้พบ เช่น ในการสกัดเอกสารอ้างอิง เมื่อทราบว่าเอกสารต้องขึ้นต้นด้วยชนิดข้อมูลนี้เท่านั้นก็สามารถระบุชนิดข้อมูลเริ่มต้นได้ทันที หรือ เมื่อข้อมูลชนิดนี้เข้ามาลำดับถัดไปต้องเป็นชนิดข้อมูลอีกอย่างหนึ่งอย่างแน่นอน เงื่อนไขเหล่านี้สามารถที่จะสร้างเป็นกฎบังคับไว้ได้เลย เมื่อข้อมูลเข้ามาก็จะถูกแบ่งชนิดข้อมูลตามกฎที่ได้กำหนดไว้ ตัวอย่าง ในการแบ่งส่วนประกอบของเอกสารอ้างอิง เช่น ชื่อผู้แต่งต้องมาก่อนชื่อเรื่อง และ ชื่อเรื่องนั้นจะประกอบไปด้วยคำตั้งแต่ 3 คำขึ้นไป ซึ่งวิธีแบบนี้มีใช้ใน Citeseer (Giles *et al*. 1998) และนำไปใช้ในงานสกัดโครงสร้างของข้อมูลในเว็บไซต์หรือไปใช้ในงานเครือข่ายคอมพิวเตอร์ในการวิเคราะห์พฤติกรรมของไวรัสคอมพิวเตอร์

ตารางที่ 1 สรุปงานวิจัยที่เกี่ยวข้องกับการสกัดเอกสารอ้างอิง

นักวิจัย	วิธีการ	ข้อดี	ข้อเสีย
C. Lee Giles *et al*. (1998)	Rule Based	พัฒนาได้ง่ายโดยอาศัยการวิเคราะห์ข้อมูลและสร้างกฎ	ไม่สามารถรองรับรูปแบบเอกสารอ้างอิงรูปแบบใหม่ๆ ได้ในทันที
Hetzner, Erik. (2008)	Learning Based	รองรับเอกสารอ้างอิงได้หลายรูปแบบ	จำเป็นต้องมีข้อมูลที่ใช้ในการสอนระบบ
Chen, Chien-Chih *et al*. (2008)	Template Based	สามารถเพิ่มเติมข้อมูลของรูปแบบโดยอาศัยมาตรฐานการเขียนเอกสารอ้างอิงรูปแบบต่างๆ	ต้องมีการทำ Template ของเอกสารอ้างอิงก่อน

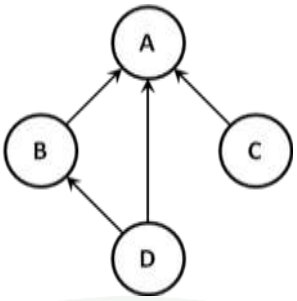
2. ความรู้เบื้องต้นเกี่ยวกับดัชนีอ้างอิง (Citation Index)

การอ้างอิงเป็นส่วนสำคัญในสังคมที่มีอยู่ในลักษณะเครือข่าย เพราะข้อมูลนั้นต้องมีการติดต่อสัมพันธ์กัน การอ้างอิงนั้นทำให้ทราบแหล่งที่มาที่ไปของข้อมูล เพราะเมื่ออ่านเนื้อหาแล้วเกิดข้อสงสัยหรือต้องการค้นคว้าเพิ่มเติม การเขียนอ้างอิงที่ท้ายบทความนั้นสามารถช่วยให้ผู้อ่านสามารถค้นคว้าเพิ่มเติมได้อย่างสะดวก

การอ้างอิงสามารถนำไปประยุกต์ใช้งานได้หลายอย่าง เช่น การค้นคืนสารสนเทศ (Information Retrieval) การค้นคืนเวปไซต์ในอัลกอริทึม PageRankนั้นอันดับของการค้นคืนต้องใช้จำนวนของการอ้างอิงร่วมด้วย โดยอันดับการค้นคืนจะขึ้นอยู่กับค่า PageRank ของเวปไซต์ที่อ้างอิงถึง ยิ่งเวปไซต์ใดถูกอ้างอิงจำนวนมากจากเวปไซต์ที่มีค่า PageRank สูง อันดับของเวปไซต์ที่ถูกอ้างอิงนั้นก็จะมีอันดับเพิ่มมากขึ้น

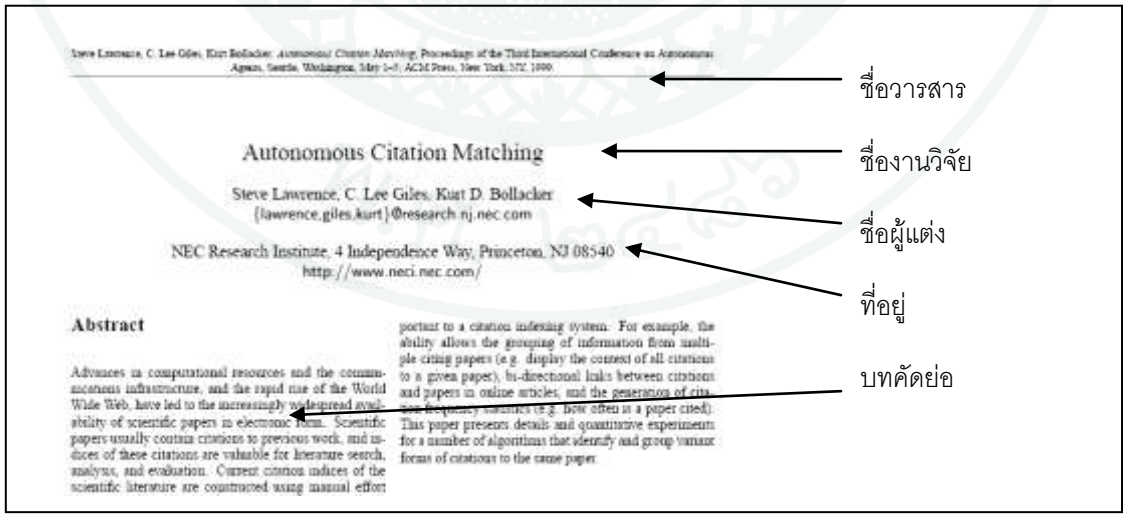
ดัชนีการอ้างอิงคือดัชนีการอ้างอิงระหว่างบทความสองบทความที่โยงใยอยู่เป็นเครือข่าย โดย การอ้างอิงเอกสารนั้นมีหลายรูปแบบ เช่น บทความ หนังสือ วารสาร หนังสือพิมพ์ เว็บไซต์ ซึ่งในการสร้างสรรค์ผลงานนั้นจำเป็นที่ต้องเขียนระบุการอ้างอิง (Citing) ก็เพื่อแสดงความน่าเชื่อถือและให้เกียรติผู้เขียนแหล่งอ้างอิงนั้น (Cited) อีกทั้งผู้อ่านสามารถค้นคว้าเพิ่มเติมจากแหล่งข้อมูลที่อ้างถึงได้ ตัวอย่าง การอ้างอิงในเว็บไซต์นั้น สามารถที่จะใช้ Hiperlink ในการอ้างถึงเนื้อหาที่ต้องการอ้างอิง ซึ่งเครือข่ายของการอ้างอิง (Citation Index) นั้นเป็นข้อมูลในรูปแบบใหม่ที่สามารถจะนำมาใช้ในการวิเคราะห์หาความสัมพันธ์ของผู้แต่งและกระแสนิยมของเนื้อหาภายในเอกสาร ซึ่งสิ่งเหล่านี้สามารถที่จะหาได้จากการทำดัชนีการอ้างอิง ข้อแตกต่างระหว่าง เอกสารอ้างอิง (references) กับ ดัชนีอ้างอิงนั้น (Citation Index) เอกสารอ้างอิงจะย้อนกลับไปหาว่างานวิจัยนี้ได้อ้างอิงเอกสารใดบ้าง แต่ดัชนีการอ้างอิงนั้นจะเป็นการดูว่างานวิจัยนี้ถูกใครอ้างอิงบ้าง

โดยงานวิจัย Citation คือ ความสัมพันธ์ระหว่างงานวิจัย 2 งาน ซึ่งได้แก่งานวิจัยที่ทำการอ้างอิงกับงานวิจัยที่ถูกอ้างอิง โดยในการทำวิจัย ผู้ทำวิจัยจะกล่าวถึงงานอื่นๆที่เป็นต้นคิดของงานวิจัยที่กำลังทำเพื่อแสดงถึงความน่าเชื่อถือและเป็นการให้เกียรติแก่เจ้าของงานดั้งเดิม เมื่อแทนการเชื่อมโยงแต่ละงานวิจัยด้วยการอ้างอิงในรูปแบบกราฟ โดยแต่ละโหนด (nodes) แทนงานวิจัย และเส้นเชื่อม (edges) แต่ละโหนดแทนการอ้างอิง โหนดแต่ละโหนดนั้นต้องไม่ซ้ำกัน และ เส้นที่เชื่อมแต่ละโหนดนั้นต้องมีเพียงเส้นเดียวและไปทางเดียวเท่านั้น ดังแสดงในภาพที่ 1



ภาพที่ 1 ตัวอย่างการอ้างอิง

ในการทำดัชนีอ้างอิง (Citation Index) ระบบจะทำการค้นหาการอ้างอิงในบทความ ได้แก่ บรรณานุกรม, เอกสารอ้างอิง หรือ เอกสารและสิ่งอ้างอิง ซึ่งข้อมูลที่ได้ก็คือรายการอ้างอิงที่มีชื่อเรื่อง ชื่อผู้แต่ง ปีที่ตีพิมพ์ และอื่นๆ จากนั้นทำการจัดเก็บข้อมูลการอ้างอิงซึ่งบางครั้งมีงานวิจัยที่อ้างถึงงานวิจัยเดียวกันแต่เขียนรูปแบบการอ้างอิงไม่เหมือนกัน ทำให้ไม่สามารถที่จะทำดัชนีอ้างอิงได้ จำเป็นต้องมีการสกัดข้อมูลออกมาจัดเก็บในรูปแบบที่อยู่ในมาตรฐานเดียวกัน เช่น BibTeX (Alexander Feder. 2006) และ EndNote (Thomson Research Soft. 1980) เพื่อใช้ในการตรวจสอบการอ้างอิงว่าเป็นการอ้างอิงงานเดียวกันหรือไม่ ประโยชน์ของดัชนีอ้างอิงก็เพื่อหาความถี่การถูกอ้างอิงของงานวิจัย งานวิจัยใดถูกอ้างอิงมากแสดงถึงความน่าเชื่อถือและความสำคัญของงานวิจัย สามารถใช้ดูว่าในปัจจุบันงานวิจัยในสาขาใดได้รับความนิยม เพื่อช่วยในการตัดสินใจทำงานวิจัยและให้เงินสนับสนุนการทำวิจัย



ภาพที่ 2 ตัวอย่างเอกสารงานวิจัย

เอกสารงานวิจัยจะมีเนื้อหาของงานแตกต่างกันไปแต่ก็จะมีรูปแบบของเอกสารคล้ายๆ กัน ดังภาพที่ 2 เป็นรูปแบบเอกสารที่ตีพิมพ์ ส่วนของการอ้างอิงนั้น ดัชนีอ้างอิงจะแปลงการอ้างอิงให้อยู่ในรูปแบบเดียวกัน ซึ่งวิธีการจัดเก็บเอกสารอ้างอิงจะมีมาตรฐานในการจัดเก็บ เช่น EndNote, BibTex ตัวอย่าง การขียนเอกสารอ้างอิง

C.L. Giles, K. Bollacker, and S. Lawrence,“CiteSeer: An Automatic Citation Indexing System,” Digital Libraries 98: Third ACM Conf. on Digital Libraries, ACM Press, New York, 1998, pp. 89-98.

สามารถเปลี่ยนเป็นรูปแบบการเก็บแบบ EndNote ได้ดังนี้

```
%0 Conference Paper
%1 276685
%A C. Lee Giles
%A Kurt D. Bollacker
%A Steve Lawrence
%T CiteSeer: an automatic citation indexing system
%B Proceedings of the third ACM conference on Digital libraries
%@ 0-89791-965-3
%C Pittsburgh, Pennsylvania, United States
%P 89-98
%D 1998
%R http://doi.acm.org/10.1145/276675.276685
%I ACM
```

หรือสามารถเปลี่ยนเป็นรูปแบบการเก็บแบบ BibTeX ได้ดังนี้

```
@INPROCEEDINGS {Giles98citeseer:an,
        author = {C. Lee Giles and Kurt D. Bollacker and Steve Lawrence},
        title = {Citeseer: an automatic citation indexing system},
        booktitle = {INTERNATIONAL CONFERENCE ON DIGITAL LIBRARIES},
        year = {1998},
        pages = {89--98},
        publisher = {ACM Press}
}
```

ซึ่งการเก็บเอกสารอ้างอิงให้อยู่ในรูปแบบมาตรฐานก็เพื่อทำให้ข้อมูลอยู่ในรูปแบบที่มีโครงสร้าง ทำให้สามารถนำไปใช้ประมวลผลได้ทันที ซึ่งมาตรฐานของ EndNote ก็เพื่อใช้โปรแกรมจัดการเอกสารอ้างอิง (Reference management system) ของ EndNote โดยมีโปรแกรมที่ใช้ในการจัดเก็บเอกสารอ้างอิงแบบโปรแกรมติดตั้งที่คอมพิวเตอร์ และที่บริการบนอินเตอร์เน็ต ได้แก่ EndNote Web โดยมีเวปไซต์ที่สนับสนุนการใช้ มาตรฐานของ EndNote เช่น ACM, IEEE และอื่นๆ

ในส่วนของการจัดเก็บเอกสารอ้างอิงมีอีกหนึ่งระบบได้แก่ BibTeX ที่เป็นระบบจัดการเอกสารอ้างอิงที่มีรูปแบบการจัดเก็บข้อมูลที่ระบุชนิดข้อมูลไว้ในรายละเอียดของเอกสารอ้างอิงโดยใช้อยู่ในโปรแกรม LaTeX (Lamport. 1994) ซึ่งในปัจจุบัน BibTeX ถูกนำไปใช้อย่างแพร่หลาย เช่น CiteSeer, ACM, IEEE และอื่นๆ

การทำดัชนีอ้างอิง (Citation Index) คือ การสร้างความสัมพันธ์ระหว่างงานวิจัย 2 งาน ที่อ้างอิงถึงกัน เพื่อใช้แสดงงานวิจัยต้นแบบและงานวิจัยที่ต่อยอดจากงานที่ทำการค้นหาอยู่ได้อย่างรวดเร็ว ซึ่งมีระบบการทำดัชนีอ้างอิง (Citation Index System) E. Garfield (1997); Steve Lawrence *et al*. (1998); Michael Ley (2002) ซึ่งแต่ละระบบมีวิธีการทำงานแตกต่างกันไป โดยแบ่งวิธีการทำงานหลักได้เป็น 2 แบบ คือ การทำดัชนีอ้างอิงด้วยมือ และการทำดัชนีแบบอัตโนมัติ (Autonomous Citation Index) ซึ่งมีข้อแตกต่างที่สำคัญของสองกลุ่มนี้ ดังรายละเอียดในตารางที่ 2

ตารางที่ 2 ความแตกต่างของการทำดัชนีด้วยมือกับแบบอัตโนมัติ

แบบทำมือ	แบบอัตโนมัติ
ทำดัชนีได้ช้ากว่าเพราะต้องใช้คนในการทำงาน	ทำดัชนีอ้างอิงได้เร็ว เนื่องจากใช้คอมพิวเตอร์ในการทำงาน
ข้อมูลไม่ทันสมัยเพราะต้องใช้เวลาในช่วงเวลาการทำงานของคน	มีการปรับปรุงข้อมูลอยู่ตลอดเวลาเมื่อมีข้อมูลเข้ามาใหม่จะทำการดัชนีโดยอัตโนมัติ
ค่าใช้จ่ายสูงเพราะต้องใช้กำลังคนในการทำงานตลอด	ค่าใช้จ่ายต่ำเพราะลงทุนเพียงครั้งเดียว
มีความถูกต้องมากกว่าเพราะคนสามารถแยกแยะ และพิจารณาข้อมูลได้ดีกว่า	มีข้อผิดพลาดในการสกัดรายการอ้างอิงอยู่บ้างทำให้ดัชนีการอ้างอิงไม่ถูกต้อง

2.1 ระบบดัชนีอ้างอิงด้วยมือ (Manual Citation Index)

ระบบการทำดัชนีอ้างอิงด้วยมือนั้นเริ่มจากสถาบัน Institute for Scientific Information (ISI) ในปี 1960 ซึ่งปัจจุบันเป็น Thomson Scientific มีระบบดัชนีอ้างอิงงานวิจัยสาขาวิทยาศาสตร์ (Science Citation Index) ที่ตีพิมพ์ในวารสารต่างๆ โดยมีขั้นตอนการทำงานเริ่มจากวารสารหรือหนังสือที่ได้ทำการตีพิมพ์ออกมาและสแกนเอกสารเพื่อเก็บข้อมูลในรูปแบบดิจิตอล จากนั้นใช้คนในการอ่านและทำดัชนีอ้างอิงระหว่างบทความ เพื่อบันทึกลงในฐานข้อมูล โดยใช้เวลาทั้งหมดประมาณ 1-2 สัปดาห์

2.2 ระบบดัชนีอ้างอิงอัตโนมัติ (Autonomous Citation Index)

ระบบดัชนีอ้างอิงอัตโนมัติถูกพัฒนาขึ้นเพื่อรองรับเอกสารงานวิจัยในปัจจุบันที่มีเป็นจำนวนมากและเกิดขึ้นใหม่ตลอดเวลา สามารถที่จะค้นหาเอกสารได้จากเว็บไซต์ ระบบการทำดัชนีการอ้างอิงอัตโนมัติ (Automatic Citation Indexing System (ACI)) สามารถช่วยรวบรวมเอกสารต่างๆโดยใช้โครงสร้างของดัชนีการอ้างอิง จุดมุ่งหมายของ ACI คือ เพื่อลดค่าใช้จ่ายในการรวบรวมเอกสาร, การจัดเก็บเอกสารที่หาได้ เพื่อเพิ่มประสิทธิภาพ และลดระยะเวลา การทำดัชนีการอ้างอิง การค้นคืนเอกสารทำได้ง่ายและรวดเร็ว ซึ่งมีงานวิจัยของ CiteSeer (Giles *et al*. 1998) ที่เป็นระบบทำดัชนีอ้างอิงอัตโนมัติโดยมีกระบวนการทำงานดังต่อไปนี้

2.2.1 Document Collection (การรวบรวมเอกสาร)

รวบรวมงานวิจัยมาทำดัชนี โดยใช้ Web crawler ไปค้นหาตามเวปเพื่อหาที่อยู่ของงานวิจัยจากนั้นจะสกัดคำต่างๆที่มีอยู่ของงานวิจัยในกลุ่มข้อความ, Usenet หรือ รายชื่ออีเมล และใช้เวบ search engine ทั่วไปเช่น AltaVista, HotBot, Excite เพื่อหางานวิจัยโดยใช้คำค้นว่า publications, papers, postscript และอื่นๆ จากนั้นทำการดาวโหลดไฟล์ที่มีนามสกุล “.pdf”, “.doc”, “.zip”, “.ps”, “ .ps.Z” หรือ “.ps.gz” โดยเก็บ URL และไฟล์ Postscript เอาไว้

2.2.2 Document Parsing (การสกัดข้อมูล)

เมื่อทำการดาวโหลดไฟล์แล้วระบบจะทำการแปลงรูปแบบไฟล์ให้อยู่ในรูปแบบข้อความ เพราะว่าในการเผยแพร่ไฟล์เอกสารงานวิจัยนั้นมีหลากหลายรูปแบบ ไฟล์ข้อความที่แปลงแล้วจะถูกตรวจสอบว่ามีอยู่ในฐานข้อมูลของการอ้างอิงหรือไม่ ถ้ายังไม่มีในฐานข้อมูลก็จะทำการสกัดข้อมูลในงานวิจัย เช่น URL, Header, Abstract, Introduction, Citations

2.2.3 Identifying Citations (การระบุตัวตนเอกสาร)

การระบุตัวตนของเอกสารนั้นคือการทำดัชนีการอ้างอิงของเอกสารเพราะว่าการเขียนเอกสารอ้างอิงนั้นมีวิธีการเขียนที่ไม่เหมือนกัน แต่เอกสารที่ระบุนั้นเป็นเอกสารอันเดียวกันจึงต้องมีการระบุตัวตนเอกสาร ดังตัวอย่าง

พรรณี ช. เจนจิต.2530.จิตวิทยาการเรียนการสอน.กรุงเทพมหานคร: บริษัทต้นอ้อ แกรมมี่ จำกัด.

พรรณี ช. เจนจิต.2538.จิตวิทยาการเรียนการสอน.กรุงเทพฯ: บริษัทต้นอ้อ แกรมมี่ จำกัด.

พรรณี ช. เจนจิต.2538.จิตวิทยาการเรียนการสอน.พิมพ์ครั้งที่ 4.กรุงเทพฯ: คอมแพคท์พริ้น จำกัด.

พรรณี ช. เจนจิต.2538.จิตวิทยาการเรียนการสอน.(พิมพ์ครั้งที่ 4).กรุงเทพมหานคร: อมรินทร์การพิมพ์.

พรรณี ชูทัย เจนจิต.2545.จิตวิทยาการเรียนการสอน.กรุงเทพมหานคร: บริษัท เมธีทิปส์ จำกัด.

3. ความรู้เบื้องต้นเกี่ยวกับการเขียนเอกสารอ้างอิง (Citation Styles)

การเขียนเอกสารอ้างอิงมีรูปแบบการเขียนแตกต่างกันไป โดยมีมาตรฐานการเขียนแบ่งตามสาขาของงานวิจัยและสถาบันที่ตีพิมพ์งานวิจัย เพื่อให้ผู้อ่านสามารถไปค้นหาเอกสารอ้างอิงเพื่อศึกษาเพิ่มเติมได้ง่าย โดยที่นิยมใช้ในปัจจุบันนั้นมีดังนี้American Psychological Association (APA) style, Council of Biology Editors (CBE) style, American Medical Association (AMA) style, Chicago style, Harvard style, Modern Languages Association (MLA) style, Turabian style, Vancouver style และ IEEE ซึ่งแต่ละรูปแบบมีการกำหนดโครงสร้างการเขียนที่แตกต่างกันในรายละเอียดแต่ก็จะมีการแบ่งประเภทของสิ่งที่อ้างอิงคล้ายๆกันคือการอ้างอิง หนังสือ, หนังสือแปล, บทความ, วารสาร, สารานุกรม, หนังสือพิมพ์, ซีดีรอม, เว็บไซต์ และ อื่นๆ

ตัวอย่างการเขียนเอกสารอ้างอิงหนังสือในมาตรฐานต่างๆ

American Psychological Association (APA) style

Okuda, M., & Okuda, D. (1993). *Star trek chronology: The history of the future*. New York, NY: Pocket Books.

Council of Biology Editors (CBE) style

Ferrini AF, Ferrini RL. Health in the later years. 2nd ed. Dubuque (IA): Brown & Benchmark; 1993. 470 p.

American Medical Association (AMA) style

Okuda M, Okuda D. *Star Trek Chronology: The History of the Future*. New York: Pocket Books; 1993.

Chicago style

Okuda, Michael, and Denise Okuda. 1993. *Star trek chronology: The history of the future*. New York: Pocket Books.

Harvard style

McCarthey, EJ, William, DP & Pascale, GQ 1997, *Basic marketing*, Irwin, Sydney.

MLA style

Okuda, Michael, and Denise Okuda. *Star Trek Chronology: The History of the Future*. New York: Pocket, 1993. Print.

Turabian style

Okuda, Michael, and Denise Okuda. 1993. *Star trek chronology: The history of the future*. New York: Pocket Books.

Vancouver style

Carlson BM. Human embryology and developmental biology. 3rd ed. St. Louis: Mosby; 2004.

IEEE style

R. Hayes, G. Pisano, D. Upton, and S. Wheelwright, *Operations, Strategy, and Technology: Pursuing the competitive edge*. Hoboken, NJ : Wiley, 2005.

4. ความรู้เบื้องต้นเกี่ยวกับอัลกอริทึมการหาค่าความเหมือนของข้อความ

การทำดัชนีอ้างอิงนั้นระบบทำการเปรียบเทียบชื่อเรื่องในเอกสารอ้างอิงกับชื่อเรื่องในฐานข้อมูลวิทยานิพนธ์ทั้งหมด โดยใช้การหาค่าความต่างของข้อความซึ่งมีอัลกอริทึมที่ใช้ในการทำงานดังต่อไปนี้

4.1 Hamming distance

Hamming distance เป็นการหาค่าความแตกต่างของสองข้อความอย่างง่าย โดยที่ข้อความทั้งสองนั้นจะต้องมีขนาดเท่ากัน อัลกอริทึมจะทำการเปรียบเทียบข้อความทั้งสองทีละตัวอักษรว่าตรงกันหรือไม่ ค่าความแตกต่างของทั้งสองข้อความก็คือจำนวนของอักษรที่แตกต่างกัน

เช่น "TONED" and "ROSES" มีค่าความแตกต่างเท่ากับ 3.

```
T  O  N  E  D
   |     |
R  O  S  E  S
```

4.2 Longest common subsequence (LCS)

Longest common subsequence คือ การหาลำดับของสายอักขระที่เหมือนกันการทำงานของอัลกอริทึมนั้นจะมีการเปรียบเทียบข้อความทีละตัวอักษรเมื่อไม่ตรงกันก็จะทำการข้ามไปเปรียบเทียบตัวอักษรถัดไป โดยมีความสามารถในการเพิ่มหรือลบตัวอักษรในข้อความ

โดยที่มีข้อความทั้งสองได้แก่ $X = (x_1, x_2, ..., x_i)$ และ $Y = (y_1, y_2, ..., y_j)$ ฟังก์ชั่นการทำงานของ LCS มีดังนี้

$$LCS(X_i, Y_j) = \begin{cases} \emptyset & \text{if } i = 0 \text{ or } j = 0 \\ (LCS(X_{i-1}, Y_{j-1}), x_i) & \text{if } x_i = y_j \\ \text{longest}(LCS(X_i, Y_{j-1}), LCS(X_{i-1}, Y_j)) & \text{if } x_i \neq y_j \end{cases}$$

เช่น X = "XMJYAUZ" และ Y = "MZJAWXU" จะได้ข้อความที่เหมือนกันระหว่าง X และ Y ได้แก่ "MJAU"

4.3 Levenshtein distance

Levenshtein distance เป็นการหาค่าความแตกต่างของข้อความสองข้อความโดยมีวิธีการทำงานคล้ายกับการหาสายอักขระที่เหมือนกันของสองข้อความแบบ Longest common subsequence (LCS) แต่เพิ่มความสามารถในการแทนที่ตัวอักษร จากเดิมที่มีความสามารถในการเพิ่มหรือลดตัวอักษรที่แตกต่างกันเท่านั้น เช่น "SUNDAY" and "SATURDAY" มีค่าความแตกต่างเท่ากับ 3 โดยมีวิธีการทำงานดังนี้

		S	A	T	U	R	D	A	Y
	0	1	2	3	4	5	6	7	8
S	1	**0**	**1**	**2**	3	4	5	6	7
U	2	1	1	2	**2**	3	4	5	6
N	3	2	2	2	3	**3**	4	5	6
D	4	3	3	3	3	4	**3**	4	5
A	5	4	3	4	4	4	4	**3**	4
Y	6	5	4	4	5	5	5	4	**3**

ตารางที่ 3 การทำงานของ Edit Distance

อัลกอริทึม	ข้อดี	ข้อเสีย
Hamming distance	มีการทำงานที่เร็วเพราะใช้การเปรียบเทียบทีละตัวอักษร	เปรียบเทียบได้เฉพาะข้อความที่มีขนาดเท่ากัน
Longest common subsequence	สามารถเปรียบเทียบข้อความที่มีขนาดไม่เท่ากันได้	มีฟีเจอร์ในการเพิ่มและลดตัวอักษรแต่ไม่มีการสับเปลี่ยนตัวอักษร
Levenshtein distance	ความสามารถเหมือนกับ LCS แต่เพิ่มความสามารถในการสับเปลี่ยนตัวอักษร	

อุปกรณ์และวิธีการ

ในหัวข้อนี้จะทำการอธิบายถึงอุปกรณ์ และวิธีการที่ใช้ในการทดลอง โดยในส่วนของอุปกรณ์จะทำการกลาวถึง ซอฟตแวรที่ใชในการทดลอง และชุดขอมูลทดสอบ และในสวนของวิธีการจะทำการกลาวถึงวิธีการของการสกัดเอกสาร

อุปกรณ์

1. ฮาร์ดแวร์ และซอฟต์แวร์ (Hardware and Software)

ฮาร์ดแวร์	Pentium 4 (1.6 GHz), Ram 1 GB
ระบบปฏิบัติการ	Microsoft Window XP
เครื่องมือพัฒนาโปรแกรม	Java SDK 1.6, Eclipse, Python 2.6.2
เครื่องมืออื่นๆ	JFlex, CUP, PDF Miner

2. ชุดข้อมูลทดสอบ (Dataset)

งานวิจัยที่เกี่ยวข้องกับการสกัดรายการเอกสารอ้างอิง (Citation Extraction) นั้นมีชุดข้อมูลทดสอบที่หลากหลาย เช่น cora (McCallum, 2005), citeseer (Bollacker *et al*., 1998) และ dblp (Ley, 2002) แต่ในวิทยานิพนธ์ฉบับนี้ทำการทดสอบกับข้อมูลเอกสารอ้างอิงที่เป็นภาษาไทย ซึ่งในงานวิจัยการสกัดเอกสารอ้างอิงภาษาไทยนั้น ยังไม่มีชุดข้อมูลทดสอบที่เป็นมาตรฐานจึงจำเป็นต้องใช้ข้อมูลทดสอบที่สร้างขึ้นมาเองสำหรับงานวิจัยนี้

ข้อมูลที่ใช้ในการทดสอบ คือเอกสารวิทยานิพนธ์ของมหาวิทยาลัยเกษตรศาสตร์ในช่วงปี 2545-2549 ที่เผยแพร่ใน http://intanin.lib.ku.ac.th โดยทำการ Download ข้อมูลเฉพาะส่วนของเอกสารอ้างอิง ซึ่งในงานวิจัยจะทำการคัดแยกเอาเฉพาะเอกสารที่เป็นภาษาไทยที่สามารถแปลงรูปแบบไฟล์ให้อยู่ในรูปแบบ Text ไฟล์ได้เท่านั้นเพราะว่าบางเอกสารผู้จัดทำได้เข้ารหัสและบางเอกสารก็เป็นการสแกนจากรูปเล่มจึงไม่สามารถนำมาใช้ในการวิจัยได้ ซึ่งในจำนวนเอกสารที่ใช้ในงานวิจัยนั้นจะทำการคัดเลือกเอกสารจำนวน 10% เพื่อใช้เป็นเอกสารทดสอบโดยเอกสารจำนวนนี้จะทำการสกัดการอ้างอิงด้วยมือเพื่อใช้ในการตรวจสอบความถูกต้อง

ตารางที่ 4 คุณสมบัติของชุดข้อมูลทดสอบ

ปีที่พิมพ์วิทยานิพนธ์*	จำนวนวิทยานิพนธ์ทั้งหมดในเว็บไซต์	จำนวนเล่มวิทยานิพนธ์ที่เขียนเป็นภาษาไทยที่ใช้ในงานวิจัย	จำนวนเล่มวิทยานิพนธ์ที่เขียนเป็นภาษาไทยที่ใช้ทดสอบ
2545	478	397	39
2546	1,690	1,433	143
2547	1,636	1,367	136
2548	1,804	1,193	119
2549	1,559	874	87
รวม	7,167	5,264	524

หมายเหตุ * ข้อมูลทั้งหมดเป็นการดาวน์โหลด ณ วันที่ 16 มีนาคม 2552

วิธีการ

วิทยานิพนธ์ฉบับนี้ เสนอวิธีการสกัดเอกสารอ้างอิงในวิทยานิพนธ์ โดยใช้วิธีการของไวยากรณ์ไม่พึ่งบริบท ในการสกัดเอกสารอ้างอิง ซึ่งในการไวยากรณ์ไม่พึ่งบริบทนั้นจะมีการเขียนกฎไวยากรณ์ของภาษา ที่ใช้ในการอธิบายรูปแบบโครงสร้างของภาษาว่าต้องประกอบไปด้วยอะไรบ้างและมีลำดับเป็นอย่างไร การเขียนไวยากรณ์ประกอบด้วยชุดของสัญลักษณ์ (G) ที่ประกอบด้วย

$$G = (T, N, R, S)$$

โดย

$$T = \{ \quad Termimal: DOT, ENDLINE, COMMA, COLON, QUOTE \quad \}$$

$$N = \left\{ \begin{array}{c} Non\ Terminal: WORD, YEAR, CITY, THESIS, UNIVERSITY, JOURNAL, \\ AUTHOR, AND, AUTHORTITLE, TITLE, PUBLISHER, \\ FACULTY, DEPARTMENT, PUBLISHING, MONTH, TYPE, \\ PROJECT, THESISB, BRANCH, ABSTRACT \end{array} \right\}$$

$$R = \{ r: \exists \omega \in (t \cup n)^* : \}$$

$$S = \{ s \in N \}$$

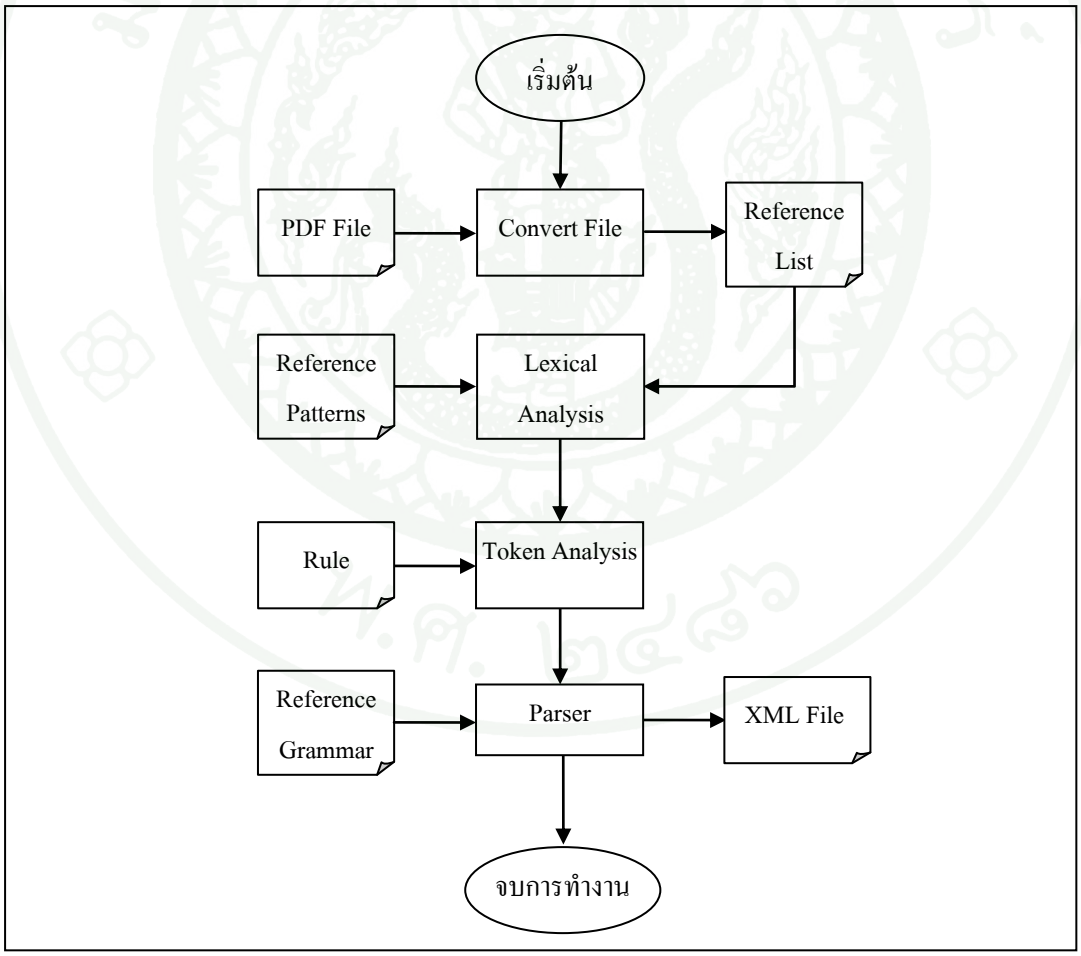
ซึ่ง Terminal คือ ชุดรายการของสัญลักษณ์ที่เป็นตัวแบ่งส่วนต่างๆของเอกสารอ้างอิง Non Terminal คือชุดของสัญลักษณ์ หรือข้อความที่มีความหมายเชิงนามธรรม และชุดของกฎหรือไวยากรณ์ภาษา (R) โดยมี S เป็นจุดเริ่มต้นของไวยากรณ์ภาษา

การใช้ Context Free Grammar (CFG) ในการสกัดเอกสารอ้างอิง ต้องทำการสรุปรูปแบบของการเขียนเอกสารอ้างอิงในวิทยานิพนธ์ที่ต้องการสกัดทั้งหมดเพื่อทำการสร้างไวยากรณ์ของเอกสารอ้างอิง ซึ่งรูปแบบการเขียนเอกสารอ้างอิงนั้นใช้มาตรฐานการเขียนเอกสารอ้างอิงของบัณฑิตวิทยาลัย โดยเป็นแบบ APA ที่เรียงตามลำดับชื่อและปี

ในการสกัดเอกสารโดยใช้ไวยากรณ์ไม่พึ่งบริบทนั้นประกอบด้วยขั้นตอนการวิเคราะห์สายอักขระ การวิเคราะห์คำ และ การจำแนกคำตามโครงสร้างของการเขียนอ้างอิงโดยใช้ Parser ซึ่งมีขั้นตอนการทำงานดังนี้

ขั้นตอนการทำงานของระบบสกัดเอกสารอ้างอิง

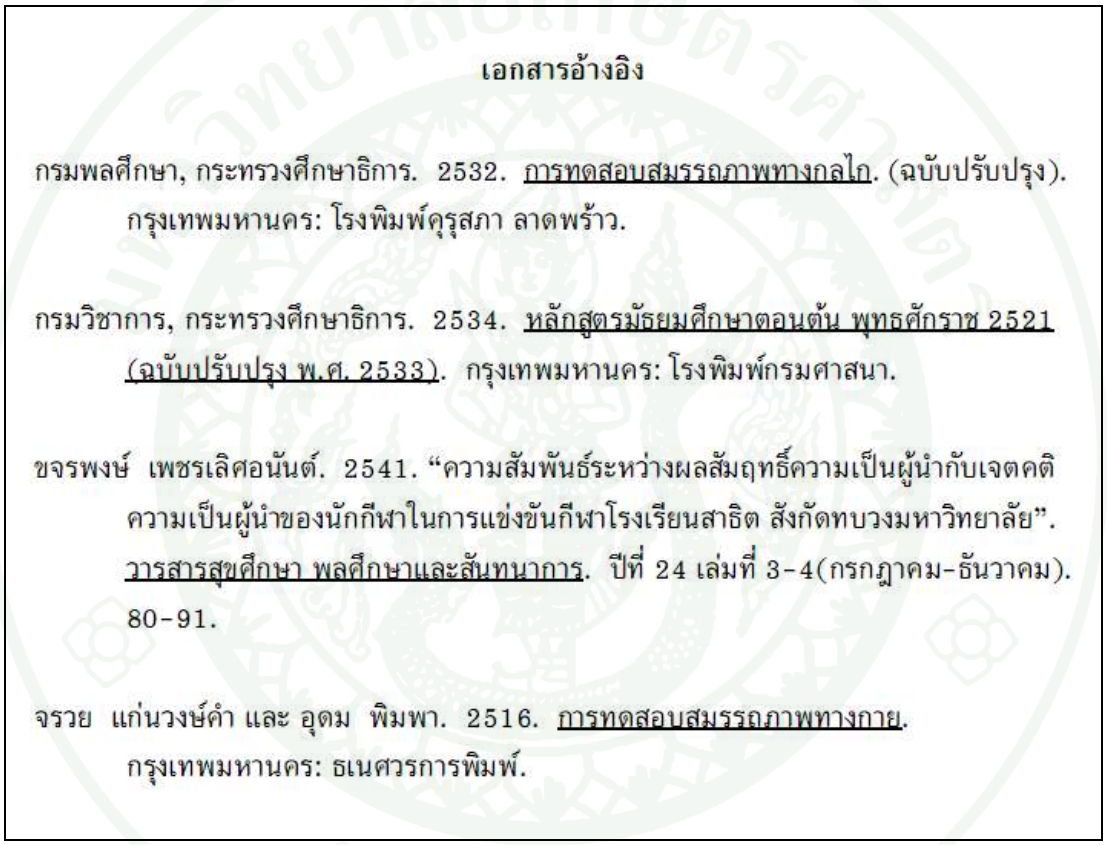
การทำงานของระบบแบ่งออกเป็นขั้นตอนหลักๆ ได้แก่ การแปลงเอกสารพีดีเอฟไฟล์เป็นเท็กไฟล์ (Convert PDF to Text), การสกัดเอกสารที่ประกอบไปด้วยการทำ Lexical Analysis, Token Analysis, Parser และ Template Classification เป็นการแบ่งประเภทของข้อมูลซึ่งระบบมีข้อมูลนำเข้าเป็นส่วนรายการอ้างอิงของวิทยานิพนธ์ที่อยู่ในรูปแบบพีดีเอฟไฟล์ (PDF) แสดงภาพรวมของระบบดังภาพที่ 3



ภาพที่ 3 การทำงานทั้งหมดของระบบสกัดเอกสารอ้างอิง

1. การแปลงรูปแบบไฟล์ (Convert File)

ส่วน PDF Convertor ในงานวิจัยได้ปรับปรุงโปรแกรม PDFminer (Yusuke Shinyama, 2004) เพื่อแปลงรหัสตัวอักษรของภาษาไทยและแก้ไขสระและวรรณยุกต์ที่ไม่ถูกต้อง โดยโปรแกรมนี้ทำหน้าที่ในการเปลี่ยนรูปแบบของไฟล์พีดีเอฟ (PDF File) ดังภาพที่ 4 ให้เป็น Reference List ที่อยู่ในรูปแบบไฟล์ข้อความ (Text File) ในภาพที่ 5



เอกสารอ้างอิง

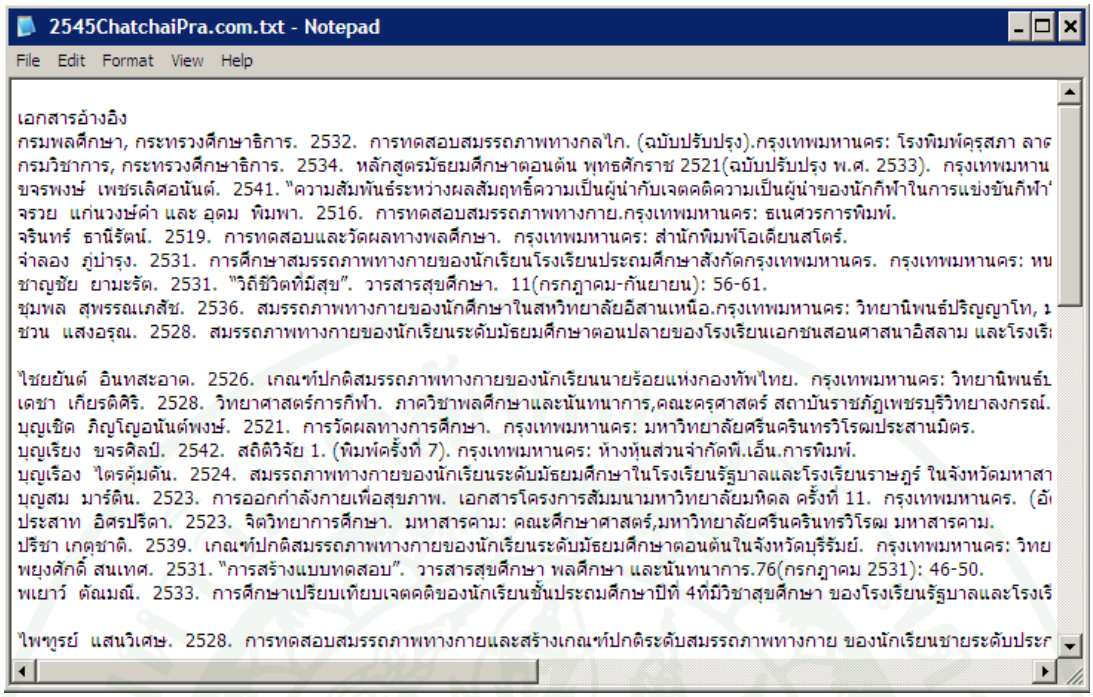
กรมพลศึกษา, กระทรวงศึกษาธิการ. 2532. การทดสอบสมรรถภาพทางกลไก. (ฉบับปรับปรุง). กรุงเทพมหานคร: โรงพิมพ์คุรุสภา ลาดพร้าว.

กรมวิชาการ, กระทรวงศึกษาธิการ. 2534. หลักสูตรมัธยมศึกษาตอนต้น พุทธศักราช 2521 (ฉบับปรับปรุง พ.ศ. 2533). กรุงเทพมหานคร: โรงพิมพ์กรมศาสนา.

ขจรพงษ์ เพชรเลิศอนันต์. 2541. "ความสัมพันธ์ระหว่างผลสัมฤทธิ์ความเป็นผู้นำกับเจตคติความเป็นผู้นำของนักกีฬาในการแข่งขันกีฬาโรงเรียนสาธิต สังกัดทบวงมหาวิทยาลัย". วารสารสุขศึกษา พลศึกษาและสันทนาการ. ปีที่ 24 เล่มที่ 3-4(กรกฎาคม-ธันวาคม). 80-91.

จรวย แก่นวงษ์คำ และ อุดม พิมพา. 2516. การทดสอบสมรรถภาพทางกาย. กรุงเทพมหานคร: ธเนศวรการพิมพ์.

ภาพที่ 4 ตัวอย่างเอกสารอ้างอิงในวิทยานิพนธ์ที่อยู่ในรูปแบบพีดีเอฟไฟล์ (PDF)



ภาพที่ 5 ตัวอย่างรายการเอกสารอ้างอิงที่ทำการแปลงให้อยู่ในรูปแบบไฟล์ข้อความ

2. การวิเคราะห์สายอักขระ (Lexical Analysis)

ในการสกัดหาส่วนของคำ ระบบใช้อักขระพิเศษในการแบ่งแยกคำ (Delimiter) และสร้างกฎเกณฑ์ของคำบางประเภทด้วยนิพจน์ปกติ (Regular Expression) เพื่อช่วยในการระบุชนิดข้อความในแต่ละส่วน และ ขึ้นบรรทัดใหม่ (End line) ที่จะเป็นตัวทำให้รู้ว่าจบการอ้างอิงแล้ว ดังตัวอย่างของการอ้างอิงวิทยานิพนธ์สามารถจะแบ่งออกเป็นส่วนๆได้ดังตารางที่ 5 โดยที่ lexeme คือข้อความที่ถูกแบ่งออกเป็นส่วนๆ Token type คือชนิดของข้อมูลที่ถูกแบ่ง และ Regular expression คือ นิพจน์ที่ใช้ในการแบ่งชนิดข้อมูล

ตารางที่ 5 การแบ่งชนิดข้อมูลโดย Lexical Analyzer

Lexeme	Token type	Regular Expression
.	Delimiter	\.
,	Delimiter	\,
:	Delimiter	\:
“	Delimiter	\”
2522	Year	([0-9]{4}-[0-9]{4})|([0-9]{4}{WhiteSpace}*-{WhiteSpace}*[0-9]{4})|([0-9]{4})
End line	End line	\r|\n|\r\n
“ ”	White Space	[\t\f]
พรรณี, ชูทัย, จิตวิทยาการเรียนการสอน, กรุงเทพมหานคร ,สำนักพิมพ์โอเดียนสโตร์	Word	[^ \r\n\t\f\.\:\,]+

พรรณี ชูทัย. 2522. จิตวิทยาการเรียนการสอน. กรุงเทพมหานคร: สำนักพิมพ์โอเดียนสโตร์.

พรรณี	ชูทัย	2522	จิตวิทยาการเรียนการสอน	กรุงเทพมหานคร	สำนักพิมพ์โอเดียนสโตร์
Word	Word	Year	Word	Word	Word .

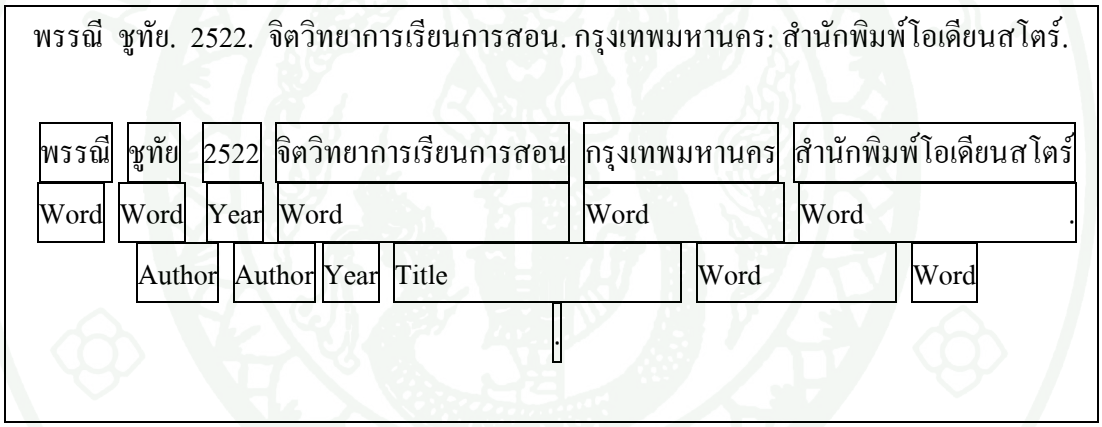
ภาพที่ 6 ตัวอย่างการทำ Lexical Analysis

ซึ่งผลที่ได้จากการทำงานส่วนนี้จะเห็นได้ว่าสายอักขระถูกแบ่งออกเป็นส่วนตามนิพจน์ที่เรากำหนดไว้ในนิพจน์ปรกติ ซึ่งแต่ละชนิดข้อมูลจะเป็นข้อความสั้น (Token) ในไวยากรณ์ของไวยากรณ์ไม่พึ่งบริบทโดยจะมีการทำงานในส่วนต่อไปเพื่อวิเคราะห์ในแต่ละ Token ว่ามีชนิดเป็นอะไรได้อีก

3. การวิเคราะห์ข้อความสั้น (Token Analysis)

ในส่วนวิเคราะห์ประเภทของข้อความนั้น ผลลัพธ์ที่ได้จาก Lexical Analyzer จะได้ชุดของข้อความ (Token) บางข้อความที่ยังไม่ทราบว่าเป็นประเภทข้อความชนิดใด จำเป็นต้องมีการจำแนก Token ออกเป็นประเภทต่างๆ เช่น ชื่อผู้แต่ง, ชื่อเรื่อง, ชื่อวารสาร, มหาวิทยาลัย, วิทยานิพนธ์ และโรงพิมพ์ เพื่อให้ Token มีประเภทข้อความที่อยู่ในไวยากรณ์ที่กำหนดไว้ ซึ่งระบบใช้วิธีการทำ Token Analyzer แบบกฎควบคู่กับคำสำคัญ

กฎในการระบุชนิดประเภทข้อความในเอกสาร คือ การแบ่งสถานะของข้อมูลออกเป็น 4 ส่วนคือ ชื่อผู้แต่ง ปีที่พิมพ์ ชื่อเรื่อง และ อื่นๆ เมื่อขึ้นต้นจะอยู่ในสถานะ ชื่อผู้แต่ง จนกว่าจะเจอ Delimiter ที่กำหนดไว้ก็จะเข้าสู่สถานะถัดไป ทำเช่นนี้ซ้ำจนกว่าเจอ End line

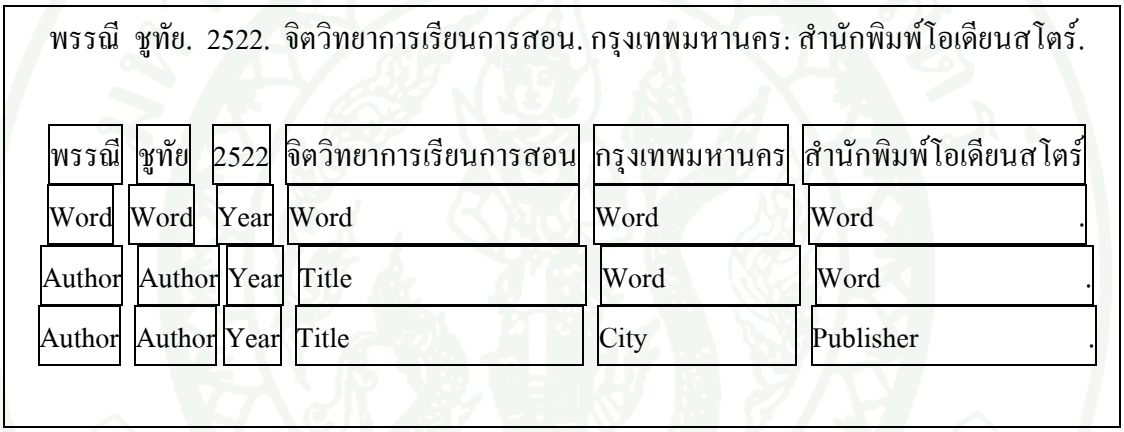


ภาพที่ 7 ตัวอย่างการทำ Token Analysis โดยใช้กฎ

คำสำคัญใช้ในการหาประเภทของชนิดข้อมูล โดยจะมีคำสำคัญคู่กับชนิดข้อมูลที่กำหนดไว้ดังตารางที่ 6 เป็นการใช้ Dictionary Search โดยเทียบข้อมูลใน Dictionary กับข้อความที่รับเข้ามาซึ่งเมื่อผ่านขั้นตอนของ Token analysis แล้วจะได้ ชุดข้อมูล Terminal ที่อยู่ใน CFG เพื่อใช้ในการทำ Parser ต่อไป

ตารางที่ 6 คำสำคัญกับชนิดข้อมูล

Keyword	Token type
และ	And
วิทยานิพนธ์	Thesis
วารสาร	Journal
มหาวิทยาลัย	University
โรงพิมพ์, สำนักพิมพ์, ...	Publisher



ภาพที่ 8 ตัวอย่างการทำ Token Analysis โดยใช้คำสำคัญ

4. การทำตัวแจง (Parser)

การทำตัวแจง (Parser) คือการนำชุดรายการของสัญลักษณ์ (Terminal) และ ข้อความที่มีความหมายเชิงนามธรรม (Non Terminal) มาเข้าไวยากรณ์โดยไวยากรณ์ ประกอบไปด้วย สัญลักษณ์เริ่มต้นที่เป็น Non Terminal กับ กฎที่มี Terminal กับ Non Terminal ผสมกัน โดยจะต้องมีการจัดเรียงตามกฎที่ได้เขียนไว้ ซึ่งครอบคลุมรูปแบบของการเขียนเอกสารอ้างอิงในวิทยานิพนธ์ที่ใช้มาตรฐานการเขียนเอกสารอ้างอิงของบัณฑิตวิทยาลัย สามารถนำมาเขียนอธิบายโดย Context Free Grammar ได้ดังต่อไปนี้

ตารางที่ 7 รายการ non terminal

refs คือ รายการเอกสารอ้างอิง ref คือ เอกสารอ้างอิง terms คือ ส่วนประกอบย่อยในเอกสารหลายส่วน term คือ ส่วนประกอบย่อยในเอกสาร detail คือ รายละเอียดปลีกย่อยของเอกสารอ้างอิง	space คือ ช่องว่าง stop คือ สัญลักษณ์ตัวคั่น authors คือ ผู้แต่งหลายคน author คือ ผู้แต่ง year คือ ปีที่พิมพ์	title คือ ชื่อเรื่อง intitle คือ ข้อความในชื่อเรื่อง words คือ ข้อความหลายข้อความต่อๆกัน ay คือ ชื่อผู้แต่งกับปีที่พิมพ์ ayt คือ ชื่อผู้แต่ง, ปีที่พิมพ์และ ชื่อเรื่อง

ตารางที่ 8 รายการ terminal

DOT คือ จุด ENDLINE คือ การขึ้นบรรทัดใหม่ COMMA คือ เครื่องหมายจุลภาค COLON คือ เครื่องหมายทวิภาค QUOTE คือ เครื่องหมายอัญประกาศ WORD คือ ข้อความที่ไม่สามารถระบุชนิดข้อมูล YEAR คือ ปีที่พิมพ์ CITY คือ จังหวัด	UNIVERSITY คือ มหาวิทยาลัย JOURNAL คือ วารสาร AUTHOR คือ ผู้แต่ง AND คือ ข้อความ "และ" AUTHORTITLE คือ คำนำหน้าชื่อ TITLE คือ ชื่อเรื่อง PUBLISHER คือ สำนักพิมพ์ FACULTY คือ คณะ DEPARTMENT คือ สาขาวิชา	PUBLISHING คือ ครั้งที่พิมพ์ MONTH คือ เดือน TYPE คือ เอกสารพิเศษ PROJECT คือ โครงการปัญหา THESISB คือ สารนิพนธ์ BRANCH คือ วิทยาเขต ABSTRACT คือ บทคัดย่อ PROCEEDING คือ งานประชุมวิชาการ THESIS คือ วิทยานิพนธ์

ตารางที่ 9 กฎของไวยากรณ์

ไวยากรณ์		คำอธิบาย
refs	::= refs ref | ENDLINE ref | ref	รายการเอกสารอ้างอิงประกอบด้วยเอกสารอ้างอิงหลายๆรายการ
ref	::= terms ENDLINE | terms words ENDLINE | terms detail ENDLINE | space	เอกสารอ้างอิงจะประกอบด้วยส่วนประกอบย่อยในเอกสารหลายๆส่วน และการขึ้นบรรทัดใหม่
terms	::= terms term | term	ส่วนประกอบย่อยในเอกสารหลายๆส่วน คือการนำส่วนประกอบย่อยหลายๆอันมาต่อกัน
term	::= words stop | YEAR stop | ayt | detail stop	ส่วนประกอบย่อย ได้แก่ ayt(ชื่อผู้แต่ง, ปีที่พิมพ์ และ ชื่อเรื่อง) หรือ ปีที่พิมพ์ตามด้วยตัวคั่น หรือ รายละเอียดปลีกย่อยของเอกสารอ้างอิงตามด้วยตัวคั่น หรือ เป็นข้อความตามด้วยตัวคั่น
ayt	::= ay words dot | ay detail:d dot | ay words COLON words dot | ay words COMMA words dot | ay title dot | ay title words dot | ay words title dot | ay words title words dot | authors words dot | authors title dot	ayt(ชื่อผู้แต่ง, ปีที่พิมพ์ และ ชื่อเรื่อง) ประกอบด้วย ชื่อผู้แต่ง, ปีที่พิมพ์ และ ชื่อเรื่อง โดยมี terminal ay(ชื่อผู้แต่ง และ ปีที่พิมพ์) และนำชื่อเรื่องมาประกอบซึ่งมีการจัดวางหลายรูปแบบ

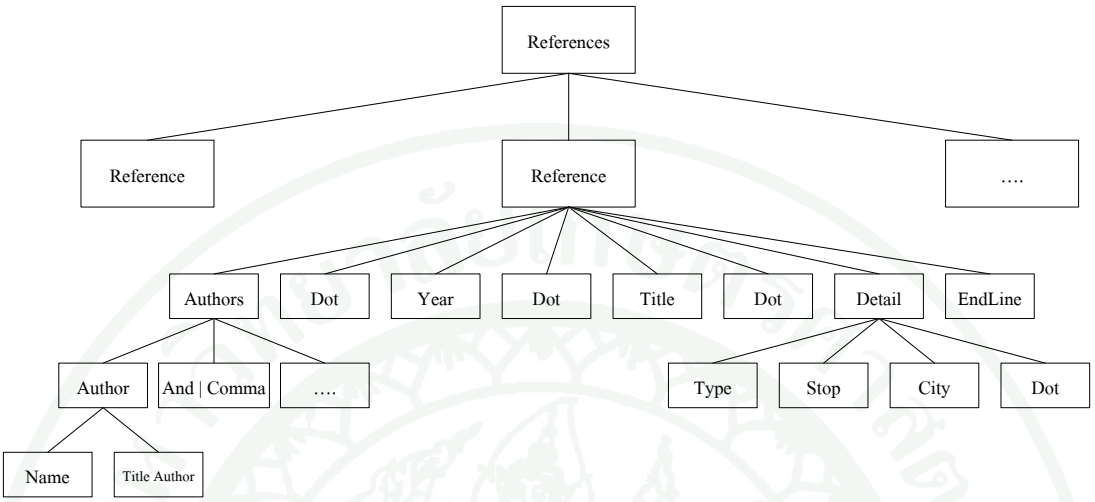
ตารางที่ 9 (ต่อ)

ไวยากรณ์		คำอธิบาย
ay	::= authors dot year | authors COMMA year | authors year | dot year	ay(ชื่อผู้แต่ง และ ปีที่พิมพ์) เป็นการประกอบกันระหว่างชื่อผู้แต่งกับปีที่พิมพ์ โดยมีตัวคั่นได้แก่ dot, COMMA หรือไม่มีตัวคั่น หรืออาจจะไม่มีผู้แต่ง
title	::= title + intitle | intitle	ชื่อเรื่องคือ title เชื่อมต่อด้วยข้อความ intitle
intitle	::= TITLE | QUOTE	intle คือ TITLE หรือ สัญลักษณ์ QUOTE
year	::= YEAR dot | YEAR	ปีที่พิมพ์ ประกอบด้วย ปีที่พิมพ์ตามด้วยตัวคั่นหรือไม่มีก็ได้
authors	::= authors AND author | authors COMMA author | authors COMMA AUTHORTITLE | authors DOT author | author | AND author | COMMA author	รายนามผู้แต่ง ประกอบไปด้วย ผู้แต่งคนด้วยหรือ หลายคน โดยมีวิธีการเขียน เชื่อมด้วยและ หรือ COMMA
author	::= author AUTHOR | AUTHORTITLE AUTHOR | AUTHOR	ผู้แต่ง ประกอบด้วย author AUTHOR หมายถึง ชื่อ และนามสกุล หรือ เป็น AUTHOR เดี่ยวๆเช่นผู้แต่งเป็นสถาบันก็จะมีแค่ข้อความเดียว

ตารางที่ 9 (ต่อ)

ไวยากรณ์		คำอธิบาย
words	::= words WORD | words YEAR | words detail | detail WORD | detail detail | WORD	ข้อความหลายข้อความประกอบด้วยข้อความที่เชื่อมต่อกันโดยไม่มีตัวคั่น
detail	::= JOURNAL | UNIVERSITY | CITY | THESIS | PUBLISHER | FACULTY | DEPARTMENT | PUBLISHING | MONTH | TYPE | PROJECT | THESISB | BRANCH | ABSTRACT	รายละเอียดปลีกย่อยของเอกสารอ้างอิง เช่น วารสาร มหาวิทยาลัย จังหวัด วิทยานิพนธ์ จำนวนครั้งที่พิมพ์ โรงพิมพ์ และ อื่นๆ
stop	::= dot | COMMA | COLON	ตัวคั่น ได้แก่ จุด จุลภาค และ_____
dot	::= DOT	เปลี่ยนจุดที่เป็น Teminal ให้เป็น Non Terminal
space	::= author ENDLINE	ส่วนที่ไม่สนใจ ได้แก่บรรทัดที่เป็นข้อความและขึ้นบรรทัดใหม่เลยโดยไม่มีตัวคั่น

โดยสามารถนำไวยากรณ์มาแสดงเป็นกราฟในรูปแบบโครงสร้างต้นไม้แบบย่อได้ดังภาพที่ 9



ภาพที่ 9 ไวยากรณ์ของรูปแบบเอกสารอ้างอิง

ซึ่งจากการทดลองนั้นยังมีข้อมูลที่ไม่สามารถจะสกัดออกมาได้ เมื่อทำการสำรวจพบว่ามีหลายสาเหตุที่ทำให้โปรแกรมไม่สามารถทำงานได้ซึ่งเกิดจากบางรายการในเอกสารอ้างอิงนั้นไม่เขียนตามกฎที่ได้กำหนดไว้ใน Context Free Grammar เช่น พิมพ์รายละเอียดไม่ครบ (ไม่ใส่จุดคั่นแต่ละส่วนของการอ้างอิง, ไม่ใส่ปีที่พิมพ์), ใส่เครื่องหมายผิด (ใส่ จุลภาค แทน จุด)

โดยการทำ Parser นั้นมีการทำงานอยู่ 2 รูปแบบได้แก่ Top down กับ Bottom up โดยงานวิจัยนี้เป็นการ Parsing Tree แบบ Bottom up ซึ่งวิธีการนั้น โปรแกรมจะนำ Token แต่ละตัวแทนโหนดต่ำสุดก่อนและจะไล่ขึ้นจนถึงโหนดรากที่เป็นรายการอ้างอิงเอกสารทั้งหมด เช่น

พรรณี ชูทัย. 2522. จิตวิทยาการเรียนการสอน. กรุงเทพมหานคร: สำนักพิมพ์โอเดียนสโตร์.

↓

Name. 2522. จิตวิทยาการเรียนการสอน. กรุงเทพมหานคร: สำนักพิมพ์โอเดียนสโตร์.

↓

Author. 2522. จิตวิทยาการเรียนการสอน. กรุงเทพมหานคร: สำนักพิมพ์โอเดียนสโตร์.

↓

Author. Year. จิตวิทยาการเรียนการสอน. กรุงเทพมหานคร: สำนักพิมพ์โอเดียนสโตร์.

↓

Author. Year. Title. กรุงเทพมหานคร: สำนักพิมพ์โอเดียนสโตร์.

↓

Author. Year. Title. City: สำนักพิมพ์โอเดียนสโตร์.

↓

Author. Year. Title. City: Publisher.

↓

Reference

↓

References

Levenshtein distance

การทำดัชนีอ้างอิงนั้นคือการยืนยันตัวตนของเอกสารในการอ้างอิงซึ่งส่วนสำคัญในการที่จะยืนยันตัวตนได้ก็คือ ชื่อเรื่อง เพราะว่าสามารถที่จะระบุการอ้างอิงได้โดยตรง ต่างจาก ชื่อผู้แต่ง และปีที่พิมพ์ที่สามารถที่จะซ้ำกันได้ เนื่องจากผู้แต่งสามารถทำงานวิจัยหลายเรื่อง แต่ชื่อผู้แต่งและปีที่พิมพ์สามารถที่ช่วยในการพิจารณาเอกสารที่ชื่อเรื่องคล้ายกันว่าเอกสารนั้นเป็นเอกสารที่ถูกอ้างอิงจริงหรือไม่ โดยขั้นตอนแรกจะทำการค้นหาชื่อผู้แต่งที่เหมือนกันจากนั้นก็จะทำการจับคู่ระหว่างชื่อเรื่อง และปีที่พิมพ์

ระบบดัชนีอ้างอิงจะทำการจับคู่ระหว่างชื่อเรื่องในฐานข้อมูลกับ ชื่อเรื่องที่อยู่ในเอกสารอ้างอิงที่สกัดออกมาได้ โดยทำการเปรียบเทียบชื่อเรื่องทั้งสอง ซึ่งถ้าผู้เขียนเอกสารอ้างอิงนั้นเขียนได้ถูกต้องตรงตามชื่อเรื่องที่ได้ทำการอ้างอิงก็จะสามารถตรวจสอบว่า ชื่อเรื่องตรงกันหรือไม่และเมื่อตรงก็สามารถทำดัชนีอ้างอิงได้ แต่จากการตรวจสอบพบว่าการเขียนเอกสารอ้างอิงของนักวิจัยนั้นไม่ตรงกับชื่อเรื่องทำให้การจับคู่นั้นไม่สามารถเปรียบเทียบชื่อเรื่องตรงๆได้ งานวิจัยนี้จึงนำเสนอการเปรียบเทียบชื่อเรื่องโดยอัลกอริธึมของ Levenshtein distance มาใช้ในการเปรียบเทียบข้อความเพื่อทำดัชนีอ้างอิง เริ่มต้นทำการสร้างเมตริกซ์ A ที่มีขนาดของเมตริกซ์กว้างคูณยาวตามขนาดของข้อความทั้งสองโดย s_1 และ s_2คือ ชื่อเรื่องที่นำมาเปรียบเทียบกัน ดังสูตร

$A[i,j] = Lev(s_1[1..i], s_2[1..j])$

$A[0,0] = 0$

$A[i,0] = i \ , \ i=1..|s_1|$

$A[0,j] = j \ , \ j=1..|s_2|$

$A[i,j] = \min(A[i-1,j-1]+\text{if } s_1[i-1]==s_2[j-1] \text{ then } 0 \text{ else } 1, A[i-1, j] + 1, A[i, j-1] + 1) \ , \ i=1..|s_1|, \ j=1..|s_2|$

จากนั้นจะได้จำนวนความแตกต่างของสายอักขระในข้อความที่ A[|s1|,|s2|] จากนั้นจะนำมาหาค่า CM คือ เปอร์เซ็นต์ความแตกต่างของสองข้อความ ดังสมการ

$$CM = A[|s_1|,|s_2|] / \max(|s_1|,|s_2|) * 100$$

โดยระบบจะกำหนดค่า threshold ไว้ที่ 20% ถ้ามีความแตกต่างกันเกินมากกว่านี้จะไม่ถือว่าเอกสารทั้งสองไม่ได้อ้างอิงถึงกันเพราะว่าข้อมูลที่สกัดได้นั้นอาจจะไม่ตรงกัน เพราะมีข้อผิดพลาดในการแปลงรูปแบบไฟล์และเพื่อป้องกันกรณีการเขียนเอกสารอ้างอิงผิดพลาดเล็กน้อยได้ด้วย

ผลและวิจารณ์

การทดลองโดยใช้ชุดข้อมูลทดสอบดังที่ได้กลาวในบทอุปกรณ์และวิธีการ ซึ่งชุดทดสอบรายการเอกสารอ้างอิงภาษาไทยนั้นใช้ เอกสารอ้างอิงของ วิทยานิพนธ์มหาวิทยาลัยเกษตรศาสตร์ ประกอบไปด้วยวิทยานิพนธ์ที่ตีพิมพ์ในช่วงปี 2545-2549 เป็นจำนวน 10% ทั้งหมด 524 ฉบับ และชุดทดสอบภาษาอังกฤษนั้นใช้รายการเอกสารอ้างอิงในวิทยานิพนธ์ที่เป็นภาษาอังกฤษทั้งหมด 500 รายการ

ตัวอย่างผลลัพธ์ของการสกัดรายการเอกสารอ้างอิง

ในการทดลองสกัดรายการเอกสารอ้างอิงได้ผลลัพธ์ที่เป็นรูปแบบ XML จะประกอบไปด้วยส่วนต่างๆของเอกสารอ้างอิงที่ถูกทำการกำกับด้วยชนิดข้อมูลดังภาพที่ 10 และภาพที่ 11

```
<REF>
        <authors>
                <author>จรวย เพชรรัตน์</author>
        </authors>
        <year>2535</year>
        <title>หลักการจัดการและบริหารธุรกิจฟาร์ม</title>
        <city>สงขลา</city>
        <faculty>ภาควิชาพัฒนาการเกษตร</faculty>
        <department>คณะทรัพยากรธรรมชาติ</department>
        <university>มหาวิทยาลัยสงขลานครินทร์</university>
</REF>
<REF>
        <authors>
                <author>เจริญ เอี่ยมสุภาษิต</author>
        </authors>
        <year>2530</year>
        <title>การจัดการฟาร์มของคณะวิชาพืชกรรมวิทยาลัยเกษตรกรรมกลุ่มภาค
ตะวันออกเฉียงเหนือกรมอาชีวศึกษา</title>
        <city>กรุงเทพมหานคร</city>
        <thesis>วิทยานิพนธ์ปริญญาโท</thesis>
        <university>มหาวิทยาลัยเกษตรศาสตร์</university>
</REF>
```

ภาพที่ 10 ผลลัพธ์การสกัดรายการเอกสารอ้างอิงภาษาไทย

```
<REF>
        <authors>
                <author>A.cau</author>,
                <author>R.kuiper</author>, and
                <author>W.-p</author>.
                <author>de roever</author>.
        </authors> <year>1992</year>.
        <title>formalising Dijkstra's development strategy within Stark's formalism</
title>. In C.B.Jones,R.C.Shaw, and T.Denvir,editors,Proc.5th.BCS-FACS Refinement.
</REF>
<REF>
        <authors>
                <author>M.kitsuregawa</author>,
                <author>H.tanaka</author>, and
                <author>T.moto-oka</author>.
        </authors> <year>1983</year>.
        < title >application of hash to data base machine and its architecture</ title >.
         New Generation Computing,1(1),
</REF>
```

ภาพที่ 11 ผลลัพธ์การสกัดรายการเอกสารอ้างอิงภาษาอังกฤษ

ผลของการทำดัชนีอ้างอิง

การทำดัชนีอ้างอิงนั้นมีวิธีการทำโดยการเปรียบเทียบเอกสารอ้างอิงในฐานข้อมูลกับรายการในเอกสารอ้างอิงที่สกัดออกมาได้ซึ่งส่วนประกอบที่นำมาใช้ในการเปรียบเทียบประกอบด้วย ชื่อผู้แต่ง ชื่อเรื่อง ปีที่พิมพ์ และแหล่งที่ตีพิมพ์ในงานวิจัยนี้ได้ทำดัชนีของวิทยานิพนธ์ของมหาวิทยาลัยเกษตรศาสตร์ โดยทำการเปรียบเทียบชื่อผู้แต่งต้องตรงกันปีที่พิมพ์ต้องเป็นปีเดียวกัน แหล่งที่ตีพิมพ์ก็ได้แก่มหาวิทยาลัยเกษตรศาสตร์ ส่วนของชื่อเรื่องนั้นไม่สามารถที่เปรียบเทียบกันโดยตรงได้เนื่องจากการเขียนชื่อเรื่องในเอกสารอ้างอิงนั้น บางครั้งเขียนไม่

เหมือนกับต้นฉบับ เพราะชื่อเรื่องวิทยานิพนธ์นั้นบางครั้งยาวทำให้อาจจะตกหล่นคำบางคำ หรือเขียนคำบางคำผิดเพี้ยนไป ดังตัวอย่างในภาพที่ 12

ทัศน์วรรณ ทองพูน. 2543. การวิเคราะห์อุปสงค์และราคาลำไยสดและผลิตภัณฑ์ลำไยของไทย
ทัศน์วรรณ ทองพูน. 2543. การวิเคราะห์อุปสงค์และราคาของลำไยสดและผลิตภัณฑ์ลำไยของไทย

ชูศรี พุทธเจริญ. 2542. บทบาทผู้บริหารโรงเรียนในการพัฒนาจริยธรรมครูในสังกัดเทศบาล
ชูศรี พุทธเจริญ. 2542. บทบาทผู้บริหารโรงเรียนในการพัฒนาจรรยาบรรณครูในสังกัดเทศบาล

วิกานดา แสนทวีสุข. 2539. รูปแบบการเรียนของนักเรียนมัธยมศึกษาตอนปลายโรงเรียนสาธิตแห่งมหาวิทยาลัยเกษตรศาสตร์
วิกานดา แสนทวีสุข. 2539. รูปแบบเรียนของนักเรียนชั้นมัธยมศึกษาตอนปลายโรงเรียนสาธิตแห่งมหาวิทยาลัยเกษตรศาสตร์

ภาพที่ 12 ตัวอย่างการเขียนชื่อเรื่องไม่ถูกต้อง

ผลลัพธ์ในการทำดัชนีอ้างอิงนั้นสามารถทำดัชนีอ้างอิงวิทยานิพนธ์ของมหาวิทยาลัยเกษตรศาสตร์เป็นจำนวนทั้งหมด 9120 เอกสาร วัดประสิทธิภาพโดยใช้ข้อมูลทดสอบจากเอกสารอ้างอิงวิทยานิพนธ์มหาวิทยาลัยเกษตรศาสตร์ 500 เอกสาร ได้ความถูกต้องที่ 95.90% ซึ่งทำการทดลองโดยการแบ่งการใช้ส่วนประกอบในการทำดัชนีเปรียบเทียบโดยตรง และเปรียบเทียบโดยการใช้อัลกอริทึม Levenshtein distance สามารถช่วยทำดัชนีในส่วนที่การเปรียบเทียบโดยตรงทำไม่ได้ถึง 39.27% ของรายการอ้างอิง ดังตารางที่ 10

ตารางที่ 10 การเปรียบเทียบชื่อผู้แต่งและชื่อเรื่องโดยอัลกอริทึม Levenshtein distance

เปรียบเทียบเฉพาะชื่อผู้แต่ง	**เปรียบเทียบเฉพาะชื่อเรื่อง**	**เปรียบเทียบชื่อผู้แต่งและชื่อเรื่องโดย Levenshtein distance**	**เปรียบเทียบชื่อผู้แต่ง, ชื่อเรื่องโดย Levenshtein distance และปีที่พิมพ์**
9796	5714	9410	9120

การวัดประสิทธิภาพ

การวัดประสิทธิภาพในงานวิจัยใช้วิธีการประเมินแบบค่าความถูกต้อง (Precision) และค่าความแม่นยำ (Recall) ซึ่งมีนิยามในการทำงานดังต่อไปนี้

โดยที่ A คือ จำนวนข้อมูลที่สกัดได้อย่างถูกต้อง
ได้แก่ ข้อมูลที่เป็นชื่อผู้แต่ง ถูกสกัดออกมาเป็นชื่อผู้แต่ง

B คือ จำนวนข้อมูลที่สกัดไม่ถูกต้องแบบสกัดขาด
ได้แก่ ข้อมูลที่เป็นชื่อผู้แต่ง แต่ถูกสกัดออกมาเป็นชนิดอื่น หรือ ไม่ถูกสกัด

C คือ จำนวนข้อมูลที่สกัดไม่ถูกต้องแบบสกัดเกิน
ได้แก่ ข้อมูลที่ไม่ใช่ชื่อผู้แต่ง แต่ถูกสกัดออกมาเป็นชื่อผู้แต่ง

D คือ จำนวนข้อมูลที่สกัดนอกเหนือจากประเภทที่ต้องการวัดประสิทธิภาพ
เช่น วัดประสิทธิภาพของการสกัดชื่อผู้แต่ง ค่า D ได้แก่ข้อมูลที่ถูกสกัดที่ไม่ใช่ผู้แต่งและถูกสกัดออกเป็นข้อมูลชนิดอื่น

1. ค่าความถูกต้อง (Token Accuracy) คือความถูกต้องในการสกัดแต่ละ Token

$$Token\ Accuracy = \frac{A + D}{A + B + C + D}$$

2. ค่าความถูกต้อง (Recall) คืออัตราส่วนของจำนวนข้อมูลที่สกัดได้อย่างถูกต้องกับจำนวนข้อมูลที่ถูกต้องใน Data Test ทั้งหมด

$$Recall = \frac{A}{A + B}$$

3. ค่าความแม่นยำ (Precision) คืออัตราส่วนของจำนวนข้อมูลที่สกัดได้อย่างถูกต้องกับจำนวนข้อมูลที่สกัดออกมาได้ทั้งหมด

$$Precision = \frac{A}{A + C}$$

4. ค่า F-Score เป็นการวัดผลจากค่า Precision กับ Recall ที่ได้มีความสมดุลเป็นอย่างไร โดยค่าที่ดีที่สุดคือ 1 และที่แย่สุดคือ 0

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

การวัดประสิทธิภาพในการสกัดเอกสารอ้างอิงนั้นใช้วิธีการเปรียบเทียบข้อมูลที่สกัดโดยโปรแกรมกับเอกสารอ้างอิงที่สกัดด้วยมือ ซึ่งข้อมูลที่นำมาใช้ในวัดประสิทธิภาพนั้น เป็นข้อมูลในส่วนของชื่อผู้แต่ง ปีที่พิมพ์และชื่อเรื่อง มาเปรียบเทียบกัน และคำนวนหา ค่าความถูกต้อง (Token Accuracy)

การวัดประสิทธิภาพของโปรแกรมนั้นใช้ส่วนของ Data test ที่กำหนดไว้เป็นจำนวน 10% ของข้อมูลที่ใช้ทำวิจัย โดยนำมาสกัดรายการอ้างอิงด้วยมือเพื่อใช้เทียบความถูกต้องของผลลัพธ์ที่ได้จากโปรแกรม ซึ่งได้ทำการแบ่งการทดสอบออกเป็น 4 ลำดับ ได้แก่ 2%, 5%, 8% และ 10% เพื่อทดสอบว่าเมื่อเพิ่มจำนวนของเอกสาร เปอร์เซ็นต์ความผิดพลาดก็จะไม่เพิ่มมากขึ้น ดังผลการวัดประสิทธิภาพในตารางที่ 11

ตารางที่ 11 ค่าความถูกต้อง (Token Accuracy)

ปี	Data Test 2%	Data Test 5%	Data Test 8%	Data Test 10%
2545	95.71%	95.79%	95.98%	95.86%
2546	96.83%	94.92%	95.28%	95.36%
2547	96.70%	95.59%	96.13%	96.44%
2548	97.24%	97.59%	97.73%	97.70%
2529	97.62%	98.32%	97.35%	97.11%

จากการวัดประสิทธิภาพของการสกัดข้อมูลที่ได้จากการสกัดก็จะได้ค่าความถูกต้องของการสกัดข้อมูลเอกสารอ้างอิง ซึ่งเป็นความถูกต้องเมื่อเทียบกับชุดข้อมูลทดสอบที่ไม่ใช่ความถูกต้องของการสกัดทั้งหมด เพราะว่าการวัดประสิทธิภาพนั้นเป็นการเปรียบเทียบกับข้อมูลที่สกัดด้วยมือซึ่งในงานวิจัยนี้ ใช้เอกสารในการทดสอบ 524 วิทยานิพนธ์ในการทดสอบซึ่งมีเอกสารอ้างอิงทั้งสิ้น 15,642 รายการ หากต้องการทราบว่าผลจากวัดประสิทธิภาพนั้น ครอบคลุมผลลัพธ์ทั้งหมดของการสกัดหรือไม่ งานวิจัยนี้จึงกำหนดช่วงเชื่อมั่น (Confidence Interval) ที่จะครอบคลุมค่า parameter อย่างน้อย 95% (100 ช่วงเชื่อมั่นจะครอบคลุมค่าจริงของความถูกต้องของการสกัดอย่างน้อย 95 ครั้ง) เพราะงานวิจัยจะไม่ทำการวัดประสิทธิภาพทั้งหมดของ Dataset จึงไม่สามารถยืนยันว่าช่วงเชื่อมั่นในการทดสอบจะครอบคลุมผลการสกัดทั้งหมด ดังนั้นจึงอาจแปรผล ได้ว่ามีความเชื่อมั่น 95% ว่าช่วงเชื่อมั่นนี้จะครอบคลุมผลของการสกัดทั้งหมด

ข้อมูลที่ใช้ในการทดสอบ (Sample size) = 15,642
สกัดข้อมูลได้ถูกต้อง = 13,504
ค่าความน่าจะเป็นในการสกัดข้อมูลได้ถูกต้อง P = 13,504 / 15,642 = 0.86

ต้องการทดสอบสมมุติฐานความน่าจะเป็นของการสกัดโดยมีค่าผิดพลาดไม่เกิน 5%
$H_0 = 0.05$

โดยกำหนดค่าความเชื่อมั่นไว้ที่ 95%
Confidence Interval (CI) = 0.95

จะได้ค่า Alpha error $\alpha = 1 -$ CI = 1 – 0.95 = 0.05

$$\mu = \frac{P - 0.05}{\sqrt{\frac{0.05 \times \alpha}{15642}}}$$

$$\mu = \frac{0.86 - 0.05}{\sqrt{\frac{0.05 \times (1 - 0.05)}{15642}}}$$

$P(\mu < 464.81)$

โดยที่ค่าความน่าจะเป็นในสกัดผิดพลาดไม่เกิน

(13,504 – 464.81) / 15,642 = 0.83 = 83%

การแยกประเภทของเอกสารนั้น มีประสิทธิภาพในการทำงานดังตารางที่ 12

ตารางที่ 12 ความถูกต้องของการแยกประเภทของเอกสาร

ปี	วารสาร	วิทยานิพนธ์	เอกสาร	หนังสือ	งานประชุมวิชาการ
2545	91.45%	99.43%	94.26%	84.65%	87.40%
2546	90.67%	99.56%	95.86%	86.03%	89.06%
2547	93.02%	99.32%	95.03%	84.87%	88.86%
2548	91.98%	99.48%	94.02%	83.45%	89.45%
2549	92.66%	99.67%	95.78%	84.36%	86.85%

การเปรียบเทียบประสิทธิภาพกับงานวิจัยก่อนหน้า

การเปรียบเทียบกับงานวิจัยก่อนหน้าเนื่องจากในการสกัดเอกสารอ้างอิงภาษาไทยนั้นยังไม่มีการทำวิจัยมาก่อนวิทยานิพนธ์ฉบับนี้จึงทำการทดลองทั้งภาษาไทยและภาษาอังกฤษ โดยใช้เอกสารอ้างอิงในวิทยานิพนธ์มหาวิทยาลัยเกษตรศาสตร์ดังที่กล่าวไว้ในส่วนอุปกรณ์และวิธีการ ซึ่งการทดลองภาษาไทยใช้วิทยานิพนธ์ 524 ฉบับ ส่วนภาษาอังกฤษใช้เอกสารอ้างอิง 500 รายการ และวัดประสิทธิภาพของการสกัดข้อมูลที่สำคัญ ได้แก่ ชื่อผู้แต่ง ชื่อเรื่อง ปีที่พิมพ์ ซึ่งในงานวิจัยนี้เป็นการสกัดเอกสารอ้างอิงด้วยวิธี Rule Based จึงทำการเปรียบเทียบกับ งานวิจัยที่เป็นวิธี Learning Based โดยใช้ Hidden Markov Model (Hetzner, 2008) กับ วิธีการแบบ Template Based (Jewell, 2002) ดังตารางที่ 13 และทดสอบกับข้อมูลภาษาไทยดังตารางที่ 14

ตารางที่ 13 การเปรียบเทียบประสิทธิภาพกับงานวิจัยก่อนหน้าด้วยชุดข้อมูลภาษาอังกฤษ

Algorithm	ผู้แต่ง			ชื่อเรื่อง			ปีที่พิมพ์		
	P	R	F1	P	R	F1	P	R	F1
HMM	93.8	97.4	95.6	89.5	90.8	90.1	83.8	91.5	87.5
Biblio Citation Parser	27.01	18.71	22.11	7.53	7.78	7.66	81.02	67.82	73.83
CFG	72.56	79.31	75.78	87.38	89.65	88.54	93.95	94.61	94.27

ตารางที่ 14 การเปรียบเทียบประสิทธิภาพกับงานวิจัยก่อนหน้าด้วยชุดข้อมูลภาษาไทย

Algorithm	ผู้แต่ง			ชื่อเรื่อง			ปีที่พิมพ์		
	P	R	F1	P	R	F1	P	R	F1
HMM	99.8	97.6	98.7	98.8	96.1	97.4	99.1	97.3	98.2
CFG	98.21	92.75	95.40	92.93	83.14	96.63	99.62	89.50	94.29

ความรู้ที่ได้จากงานวิจัย

ผลที่ได้จากการทำวิจัยนั้นสามารถวิเคราะห์การอ้างอิงของเอกสารในวิทยานิพนธ์ของมหาวิทยาลัยเกษตรศาสตร์ ว่ามีประเภทการอ้างอิงที่หลากหลาย และ อัตราส่วนระหว่างการอ้างอิงเอกสารที่เป็นภาษาไทยกับภาษาอังกฤษเป็นอย่างไร ซึ่งข้อมูลในตารางที่ 15 จะพบว่าเอกสารอ้างอิงนั้นส่วนมากจะเป็นการอ้างอิงเอกสารภาษาไทย โดยมีอัตราส่วนการอ้างอิงภาษาไทยทั้งหมด 58.05% และภาษาอังกฤษที่ 41.94% แต่แนวโน้มของการอ้างอิงเอกสารอ้างอิงภาษาอังกฤษนั้นมีอัตราการอ้างอิงเพิ่มสูงขึ้น

ตารางที่ 15 จำนวนเอกสาร และจำนวนการอ้างอิง

ปี	จำนวนเล่ม	อ้างอิงเอกสาร ภาษาอังกฤษ	อ้างอิงเอกสาร ภาษาไทย	รวม	เฉลี่ยการ อ้างอิงต่อเล่ม
2545	397	7,745 (40.24%)	11,504 (59.76%)	19,249	48.49
2546	1433	30,035 (41.87%)	41,702 (58.13%)	71,737	50.06
2547	1367	28,179 (41.97%)	38,954 (58.03%)	67,133	49.11
2548	1193	22,924 (41.44%)	32,398 (58.56%)	55,322	46.37
2549	874	18,422 (44.22%)	23,240 (55.78%)	41,662	47.67

ชนิดของเอกสารอ้างอิงที่สกัดออกมาได้นั้นยังสามารถนำมาอธิบายถึงคุณภาพของวิทยานิพนธ์ได้ว่า ในงานต่างๆนั้น ผู้ที่ทำงานวิจัยได้ค้นคว้าหาความรู้อย่างครอบคลุมหรือไม่ เพราะเนื้อหาในเอกสารแต่ละชนิดมีคุณสมบัติของข้อมูลไม่เหมือนกัน เช่น หนังสือเป็นแหล่งข้อมูลที่มีการรวบรวมเนื้อหาอย่างครบถ้วนแต่ความสดใหม่ข้อมูลนั้นอาจจะช้ากว่าวารสาร เพราะวารสารที่เป็นการรวบรวมงานวิจัยที่เพิ่งวิจัยคิดค้นขึ้นมาใหม่ แต่ข้อมูลจากวารสารบางครั้งนั้นช้ากว่า เอกสารจากงานประชุมวิชาการ เพราะนักวิจัยจะทำการตีพิมพ์ในงานประชุมวิชาการ แล้วทำเพิ่มเติมก่อนที่จะส่งตีพิมพ์ที่วารสารก็ได้ โดยผลลัพธ์ของชนิดการอ้างอิงเอกสารแสดงในตารางที่ 16

ตารางที่ 16 ประเภทเอกสารที่ถูกอ้างอิง

ปี	อ้างอิงเอกสารภาษาไทย	วารสาร	วิทยานิพนธ์ปริญญาโท	วิทยานิพนธ์ปริญญาเอก	หนังสือ	เอกสารงานประชุมวิชาการ	อื่นๆ
2545	11,504	417	2,667	35	3,120	353	35	5,937
		(3.62%)	(23.18%)	(0.30%)	(27.12%)	(3.07%)	(0.30%)	(42.39%)
2546	41,702	1,441	10,212	144	10,646	1,390	200	21,228
		(3.46%)	(24.49%)	(0.35%)	(25.53%)	(3.33%)	(0.48%)	(42.37%)
2547	38,954	1,303	9,863	134	10,147	1,422	174	19,304
		(3.34%)	(25.32%)	(0.34%)	(26.05%)	(3.65%)	(0.45%)	(40.85%)
2548	32,398	1,202	7,722	90	8,275	1,050	172	16,743
		(3.71%)	(23.83%)	(0.28%)	(25.54%)	(3.24%)	(0.53%)	(42.86%)
2549	23,240	925	6,040	102	5,751	736	98	11,679
		(3.98%)	(25.99%)	(0.44%)	(24.75%)	(3.17%)	(0.42%)	(41.26%)

เมื่อนำผลการทดลองเฉพาะการอ้างอิงวิทยานิพนธ์นั้นมาวิเคราะห์ โดยทำการจัดอันดับการอ้างอิงเรียงตามจำนวนการอ้างอิงวิทยานิพนธ์ของมหาวิทยาลัยต่างๆ จะสังเกตได้ว่า มีการอ้างอิงมหาวิทยาลัยเกษตรศาสตร์เป็นอันดับหนึ่ง ซึ่งจำนวนที่สูงนั้นเพราะวิทยานิพนธ์ที่นำมาวิจัยเป็นของมหาวิทยาลัยเกษตรศาสตร์ ทำให้จำนวนการอ้างอิงตนเองมีจำนวนสูง ซึ่งส่วนมากที่เป็นการอ้างอิงตนเองนั้นก็เพราะเป็นการทำงานวิจัยที่ต่อเนื่อง เป็นการพัฒนางานเดิม ดังตารางที่ 17 และตารางที่ 18 เป็นการแสดงอันดับของวารสารที่ถูกอ้างอิงเรียงตามจำนวนครั้งที่ถูกอ้างอิง

ตารางที่ 17 อันดับมหาวิทยาลัยที่ถูกอ้างอิงแต่ละปี

	มหาวิทยาลัยที่ถูกอ้างอิง				
อันดับ	**ปี 2545**	**ปี 2546**	**ปี 2547**	**ปี 2548**	**ปี 2549**
1	ม.เกษตร (53.45%)	ม.เกษตร (54.38%)	ม.เกษตร (55.00%)	ม.เกษตร (56.40%)	ม.เกษตร (57.59%)
2	ม.จุฬาฯ (14.15%)	ม.จุฬาฯ (11.76%)	ม.จุฬาฯ (12.00%)	ม.จุฬาฯ (11.04%)	ม.จุฬาฯ (11.17%)
3	ม.มหิดล (6.44%)	ม.มหิดล (8.43%)	ม.มหิดล (7.63%)	ม.มหิดล (7.06%)	ม.มหิดล (6.00%)

ตารางที่ 18 อันดับวารสารที่ถูกอ้างอิงแต่ละปี

	วารสารที่ถูกอ้างอิง				
อันดับ	**ปี 2545**	**ปี 2546**	**ปี 2547**	**ปี 2548**	**ปี 2549**
1	รายงานเศรษฐกิจรายเดือน (5.72%)	วารสารวิชาการ (4.71%)	วารสารวิชาการ (5.69%)	วารสารวิชาการ (8.71%)	วารสารการศึกษา (4.28%)
2	วารสารวิชาการ (4.72%)	วารสารสุขศึกษา (2.39%)	วารสารสุขศึกษา (2.84%)	วารสารการศึกษา (3.79%)	วารสารสุขศึกษา (2.54%)
3	วารสารสุขศึกษา (4.47%)	วารสารการประมง (2.39%)	วารสารสหกรณ์ (2.68%)	วารสารสหกรณ์ (2.58%)	วารสารอาหาร (2.43%)

สรุปผลการวิจัย และข้อเสนอแนะ

วิทยานิพนธฉบับนี้เสนอการวิจัยสกัดรายการเอกสารอ้างอิงทั้งภาษาไทย กับภาษาอังกฤษ ซึ่งสามารถสรุปผลการวิจัย และขอเสนอแนะสำหรับการวิจัยตอไปในอนาคต

สรุปผลการวิจัย

วิทยานิพนธฉบับนี้ไดเสนอระบบสกัดเอกสารอ้างอิงโดยใช้ Context Free Grammar สกัดรายการเอกสารอ้างอิงของวิทยานิพนธ์มหาวิทยาลัยเกษตรศาสตร์ ในการทดลองสกัดเอกสารนั้น ระบบจะแยกออกเป็นสองแบบได้แก่ เอกสารอ้างอิงภาษาไทย และ เอกสารอ้างอิงภาษาอังกฤษ เพราะรูปแบบการเขียนเอกสารอ้างอิงไม่เหมือนกัน โดยเฉพาะการเขียนชื่อผู้แต่ง เนื่องจากในการเขียนชื่อผู้แต่งในเอกสารอ้างอิงภาษาไทยที่ใช้ชื่อ และนามสกุล แบบเต็ม ไม่เหมือนการเขียนชื่อผู้แต่งในเอกสารอ้างอิงภาษาอังกฤษนั้น มีการใช้คำย่อ ทำให้รูปแบบการเขียนชื่อผู้แต่งนั้นมีหลากหลายรูปแบบกว่าภาษาไทย

ส่วนความแตกต่างระหว่างภาษาไทยกับภาษาอังกฤษนั้นการเขียนประโยคจะติดกัน ไม่มีการแยกคำในประโยค ทำให้การวิเคราะห์ส่วนประกอบทำได้ยากกว่า เพราะรายละเอียดในแต่ละส่วนของเอกสารอ้างอิงนั้นภาษาไทยจะมีจำนวน Token น้อยกว่า ทำให้การสกัดทำได้ยากซึ่งในงานวิจัยนี้ใช้ Rule Based ในการสกัดเอกสารอ้างอิง ซึ่งจะสามารถที่จะวิเคราะห์แต่ล่ะส่วนของ Token ได้ดีกว่า Learning Based กับ Template Based

ผลการทดลองที่ได้มีส่วนที่สกัดเอกสารอ้างอิงไม่ได้เนื่องจากมีการเขียนเอกสารอ้างอิงผิดรูปแบบ เช่น พิมพ์ส่วนประกอบของเอกสารอ้างอิงไม่ครบ ได้แก่ ไม่ใส่ปีที่พิมพ์ ไม่ใส่สัญลักษณ์ที่ใช้ในการคั่นรายละเอียดของเอกสารอ้างอิงที่กำหนดไว้ในมาตรฐาน ดังตัวอย่าง

เพ็ญแข แสงแก้ว.การวิจัยทางสังคมศาสตร์.ภาควิชาคณิตศาสตร์และสถิติ คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยธรรมศาสตร์, กรุงเทพฯ.252 น.
อรวินทร์ วงศ์มีเกียร.2527.การผลิตเอนไซม์บรอมิเลนจากส่วนเหลือทิ้งของสับปะรด กรุงเทพมหานคร วิทยานิพนธ์ปริญญาโท มหาวิทยาลัยเกษตรศาสตร์

สำหรับประสิทธิภาพโดยรวมของระบบสามารถสกัดรายการเอกสารอ้างอิงมีความถูกต้อง 94.22% และสามารถระบุประเภทของเอกสารที่ถูกอ้างอิงมีความถูกต้อง 92.77% จากวิทยานิพนธ์ที่ใช้ในการทดสอบจำนวน 524 ฉบับ

งานวิจัยนี้ได้ทดลองสกัดเอกสารอ้างอิงของวิทยานิพนธ์มหาวิทยาลัยเกษตรศาสตร์ เพื่อนำข้อมูลการอ้างอิงไปวิเคราะห์หารูปแบบการอ้างอิงวิทยานิพนธ์ที่เกิดขึ้น ซึ่งผลลัพธ์ของการอ้างอิงนั้นสามารถที่จะนำไปวิเคราะห์ข้อมูลต่างๆได้อย่างมากมาย เช่นหารูปแบบการอ้างอิง สาขางานวิจัยที่เป็นที่นิยมในปัจจุบัน หรือการจัดอันดับงานวิจัย, ผู้ทำงานวิจัย หรือ สังกัดที่งานวิจัยตีพิมพ์

ระบบการทำดัชนีอ้างอิงอัตโนมัติ เมื่อทำดัชนีอ้างอิงเสร็จสิ้นเราจะได้ข้อมูลที่ช่วยค้นคืนเอกสารที่แสดงข้อมูลการถูกอ้างอิงได้ว่าวิทยานิพนธ์ฉบับนี้ได้มีใครทำการอ้างอิงบ้าง มีจำนวนการถูกอ้างอิงเป็นจำนวนเท่าไร เพื่อใช้ใน การคัดเลือกวิทยานิพนธ์ที่ดีเด่น หรือให้รางวัลแก่นักวิจัย

ข้อเสนอแนะ

งานวิจัยนี้ได้ทดลองสกัดเอกสารอ้างอิงของวิทยานิพนธ์มหาวิทยาลัยเกษตรศาสตร์ เพื่อนำข้อมูลการอ้างอิงไปวิเคราะห์หารูปแบบการอ้างอิงวิทยานิพนธ์ที่เกิดขึ้น โดยใช้ระบบการทำดัชนีอ้างอิงอัตโนมัติ ซึ่งผลลัพธ์ของการอ้างอิงนั้นสามารถที่จะนำไปวิเคราะห์ข้อมูลต่างๆได้อย่างมากมาย เช่นหารูปแบบการอ้างอิง สาขางานวิจัยที่เป็นที่นิยมในปัจจุบัน หรือการจัดอันดับงานวิจัย, ผู้ทำงานวิจัย, สังกัดที่งานวิจัยตีพิมพ์

อุปสรรคในการทำงานวิจัยของการสกัดเอกสารอ้างอิงในวิทยานิพนธ์ภาษาไทย คือ ชุดทดสอบในการวัดประสิทธิภาพของระบบนั้นไม่มีชุดข้อมูลทดสอบที่มีมาตรฐาน ทำให้ผู้วิจัยต้องจัดทำชุดข้อมูลทดสอบขึ้นเอง ซึ่งแตกต่างจากภาษาอังกฤษที่มีชุดข้อมูลทดสอบมาตรฐาน เช่น CORA Dataset และอุปสรรคในการเปรียบเทียบประสิทธิผลของอัลกอริทึมนั้น คือ ผลลัพธ์ที่ได้จากโปรแกรมของ HMM (Hetzner, 2008) นั้นเป็นค่าที่วัดประสิทธิของโปรแกรม ซึ่งนำมาเปรียบเทียบได้ในระดับหนึ่งเท่านั้น และโปรแกรมของParaCite (Jewell, 2002) นั้นไม่รองรับเอกสารภาษาไทย

แนวทางวิจัยในอนาคตคือแก้ข้อผิดพลาดจากการสกัดข้อมูลจากไฟล์ที่แปลงมาจากพีดีเอฟ พัฒนาอัลกอริธึมในการสกัดเอกสารอ้างอิงให้มีความแม่นยำมากขึ้นและสามารถสกัดการอ้างอิง

จากแหล่งอื่นๆ ที่ไม่ใช่การอ้างอิงของวิทยานิพนธ์ และรวบรวมวิทยานิพนธ์ทั้งหมดที่เป็นภาษาไทยของมหาวิทยาลัยอื่นๆนอกเหนือจากวิทยานิพนธ์มหาวิทยาลัยเกษตรศาสตร์

งานวิจัยนี้สามารถนำมาประยุกต์ใช้ในการตรวจสอบความถูกต้องของการเขียนเอกสารอ้างอิงเหมือนกับ โปรแกรมจัดเก็บข้อมูลของเอกสารอ้างอิง เช่น EndNote โดยเพิ่มรูปแบบของการเขียน เอกสารอ้างอิงแบบต่างๆ และเพิ่มฐานข้อมูลในการวิเคราะห์การอ้างอิงเอกสารต่างๆ ในรูปแบบภาษาไทย เช่น รายการอ้างอิงของหนังสือ รายการอ้างอิงจากงานประชุมวิชาการ และเอกสารออนไลน์ที่มีการใส่รายการอ้างอิง

เอกสารและสิ่งอ้างอิง

บัณฑิตวิทยาลัย. **คู่มือวิทยานิพนธ์ สายวิทยาศาสตร์ ปี 2553.** แหล่งที่มา: http://www.grad.ku.ac.th/thesis/manual_sc.php, 10 กุมภาพันธ์ 2552.

พิชัย กิตติคง และ ชุลีรัตน์ จรัสกุลชัย. 2551. การวิเคราะห์การอ้างอิงของวิทยานิพนธ์ในมหาวิทยาลัยเกษตรศาสตร์, *ใน* **The 5th International Joint Conference on Computer Science and Software Engineering (JCSSE 2008)** . Kanchanaburi, Thailand.

พิชัย กิตติคง และ ชุลีรัตน์ จรัสกุลชัย. 2552. การสกัดการอ้างอิงของวิทยานิพนธ์ด้วย Context Free Grammar, *ใน* **National Computer Science and Engineering Conference (NCSEC 2009)** . Bangkok, Thailand.

พูลสุข เอกไทยเจริญ. 2551. **การเขียนรายงานการค้นคว้า : พร้อมตัวอย่างรายงานและภาคนิพนธ์.** สุวีริยาสาส์น, กรุงเทพฯ.

ศูนย์ดัชนีการอ้างอิงวารสารไทย. **Thai-Journal Citation Index Centre.** แหล่งที่มา: http://www.kmutt.ac.th/jif/public_html/index.html, 10 มิถุนายน 2552.

Alexander Feder. 2006. **BIBTEX.** Available Source: http://bibtex.org/, December 20, 2009.

Andrew McCallum. 2005. **Andrew Mccallum's Code and Data.** Available Source: http://www.cs.umass.edu/~mccallum/code-data.html, December 20, 2009.

Kurt D. Bollacker, Steve Lawrence and C. Lee Giles. 1998. CiteSeer: an Autonomous Web agent for automatic retrieval and identification of interesting publications, pp. 116 - 123. *In* **Proceedings of the second international conference on Autonomous agents** . ACM, New York, NY, USA.

Chen, Chien-Chih, Kai-Hsiang Yang, Hung-Yu Kao and Jan-Ming Ho. 2008. BibPro: A Citation Parser Based on Sequence Alignment Techniques, pp. 1175-1180. *In* **Proceedings of the 22nd International Conference on Advanced Information Networking and Applications - Workshops** . IEEE Computer Society, Washington, DC, USA.

Chris Shoemaker. 2009. **FreeCite - An Open Source Free-Text Citation Parser.** FreeCite. Available Source: http://freecite.library.brown.edu/, December 28, 2009.

C. Lee Giles, Kurt D. Bollacker and Steve Lawrence. 1998. CiteSeer: An Automatic Citation Indexing System, p. 89-98. *In* **Third ACM Conference on Digital Libraries** . ACM Press, New York.

Cortez, Eli, Altigran S. da Silva, Marcos André Gonçalves, Filipe Mesquita and Edleno S. de Moura. 2007. FLUX-CiM: Flexible Unsupervised Extraction of Citation Metadata, pp. 215 - 224. *In* **Proceedings of the 7th ACM/IEEE-CS joint conference on Digital libraries** . ACM, Vancouver, BC, Canada.

Culotta, Aron, Ron Bekkerman and Andrew McCallum. 2004. Extracting social networks and contact information from email and the web, *In* **First Conference on Email and Anti-Spam (CEAS)** . Mountain View, CA.

Ding, Ying, Gobinda Chowdhury and Schubert Foo. 1999. Template Mining for the Extraction of Citation From Digital Documents, pp. 47-62. *In* **Proceedings of the Second Asian Digital Library Conference** .

Elliotte Rusty, Harold. 1999. **XML bible.** IDG Books Worldwide, Foster City, Calif.

E Garfield. 1997. **Citation Indexing It's Theory and Application in Science, Technology, and Humanities.** Wiley, New York.

Ethem Alpaydin. 2004. **Introduction to machine learning.** MIT Press, Cambridge, Mass.

Fukuda K, Tamura A, Tsunoda T and Takagi T. 1998. Toward information extraction: identifying protein names from biological papers, *In* **The Pacific Symposium on Biocomputing (PSB)** .

Han, Hui, C. Lee Giles, Eren Manavoglu, Hongyuan Zha, Zhenyue Zhang and Edward A. Fox. 2003. Automatic document metadata extraction using support vector machines, pp. 37 - 48. *In* **Proceedings of the 3rd ACM/IEEE-CS joint conference on Digital libraries** . IEEE Computer Society, Washington, DC, USA.

Hetzner, Erik. 2008. A simple method for citation metadata extraction using hidden markov models, pp. 280-284. *In* **Proceedings of the 8th ACM/IEEE-CS joint conference on Digital libraries** . ACM, New York, NY, USA.

Isaac G. Councill, C. Lee Giles and Min-Yen Kan. 2008. ParsCit: An open-source CRF reference string parsing package, *In* **Proceedings of the Language Resources and Evaluation Conference (LREC 08)** . Marrakesh, Morrocco.

Kan, Min-Yen. 2004. **ParsCit: An open-source CRF Reference String Parsing Package.** ParsCit. Available Source: http://wing.comp.nus.edu.sg/parsCit/, November 26, 2009.

Steve Lawrence, C. Lee Giles and Kurt D. Bollacker. 1999. Autonomous citation matching, pp. 392 - 393. *In* **Proceedings of the third annual conference on Autonomous Agents** . ACM, New York, NY, USA.

Steve Lawrence, C. Lee Giles and Kurt Bollacker. 1999. Digital Libraries and Autonomous Citation Indexing. **Computer** 32 (6): 67 - 71.

Leslie Lamport. 1994. **LATEX: A Document Preparation System.** Addison Wiley.

Andrew McCallum and Wei Li. 2003. Early results for named entity recognition with conditional random fields, feature induction and web-enhanced lexicons, pp. 188-191. *In* **Human Language Technology Conference** . Association for Computational Linguistics, Morristown, NJ, USA.

Michael Ley. 2002. The DBLP Computer Science Bibliography: Evolution, Research Issues, Perspectives, pp. 1 - 10. *In* **Proceedings of the 9th International Symposium on String Processing and Information Retrieval** . Springer-Verlag, London, UK.

Mike Jewell. 2002. **ParaCite.** Available Source: http://paracite.eprints.org/, December 20, 2009.

Fuchun Peng and Andrew McCallum. 2006. Information extraction from research papers using conditional random fields. **Information Processing and Management: an International Journal** 2006 (42): 963 - 979.

Yusuke Shinyama. 2004. **PDFMiner.** Available Source: http://www.unixuser.org/~euske/python/pdfminer, November 16, 2009.

David Pinto, Andrew McCallum, Xing Wei and W. Bruce Croft. 2003. Table extraction using conditional random fields, pp. 235-242. *In* **Annual ACM Conference on Research and Development in Information Retrieval** . ACM, New York, NY, USA.

Rabiner, L.R. 1989. A tutorial on hidden Markov models and selected applications in speech recognition. **Proceedings of the IEEE** 77 (2): 257-286.

Riloff E., and Lehnert, W. 1993. Automated Dictionary Construction for Information Extraction from Text, *In* **Proceedings of the Ninth IEEE Conference on Artificial Intelligence for Applications** . IEEE, 93-99.

Seymore, K., Mccallum, A. K., and Rosenfeld, R. 1999. Learning hidden markov model structure for information extraction, pp. 37-42. *In* **AAAI 99 Workshop on Machine Learning for Information Extraction** . Orlando, Florida.

Shutao Li, James T. Kwok, Hailong Zhua, Yaonan Wang. 2003. Texture classification using the support vector machines. **Pattern Recognition** 36 (12): 2883-2893.

Koichi Takeuchi and Nigel Collier. 2002. Use of support vector machines in extended named entity recognition, pp. 1-7. *In* **proceedings of the 6th conference on Natural language learning** . Association for Computational Linguistics, Morristown, NJ, USA.

Thomson Research Soft. 1980. **EndNote.** Available Source: http://www.endnote.com, December 20, 2009.

Paul Viola and Mukund Narasimhan. 2005. Learning to extract information from semi-structured text using a discriminative context free grammar, pp. 330 - 337. *In* **Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval** . ACM, New York, NY, USA.

Alvin W. Drake. 1967. **Fundamentals of Applied Probability Theory.** McGraw-Hill, New York.

William W. Cohen and Sunita Sarawagi. 2004. Exploiting dictionaries in named entity extraction: combining semi-Markov extraction processes and data integration methods, pp. 89-98. *In* **International Conference on Knowledge Discovery and Data Mining** . ACM, New York, NY, USA.

ภาคผนวก

ภาคผนวก ก

ตัวอย่างข้อมูล

ตัวอย่างข้อมูลสอนระบบ (Training Data) ของโปรแกรม HMM

```
<!-- <s:NAME[[:space:]]+CLASS="AUTHOR">\([^<]+\)</s:NAME>
<s:AUTHOR>\1</s:AUTHOR> -->
<entries  xmlns:s="http://example.org/state"  xmlns:sy="http://example.org/style">

<entry><s:AUTHOR>กิติมา ปรีดีดิลก</s:AUTHOR>.<s:DATE>2532</s:DATE>.<s:TITLE>
การบริหารและการนิเทศการศึกษาเบื้องต้น</s:TITLE>.<s:LOCATION>กรุงเทพมหานคร
</s:LOCATION>:<s:PUBLISHER>อักษรการพิมพ์</s:PUBLISHER>.</entry>

<entry><s:AUTHOR>จินตนา เทพพิทักษ์</s:AUTHOR>.<s:DATE>2540
</s:DATE>.<s:TITLE>สภาพปัจจุบันและปัญหาเกี่ยวกับการวางแผนบริหารโรงเรียนประถมศึกษา
ในจังหวัดฉะเชิงเทรา</s:TITLE>.<s:LOCATION>กรุงเทพมหานคร</s:LOCATION>: <thesis>
วิทยานิพนธ์ปริญญาโท</thesis>,<s:INSTITUTION>มหาวิทยาลัยเกษตรศาสตร์
</s:INSTITUTION>.</entry>

<entry><s:AUTHOR>ธีระวุฒิ ประทุมนพรัตน์</s:AUTHOR>.<s:DATE>2529
</s:DATE>.<s:TITLE>การบริหารและการนิเทศการศึกษาเบื้องต้น</s:TITLE>.<s:LOCATION>
สงขลา</s:LOCATION>:<s:INSTITUTION>คณะศึกษาศาสตร์</s:INSTITUTION>,
<s:INSTITUTION>มหาวิทยาลัยศรีนครินทรวิโรฒ สงขลา</s:INSTITUTION>.</entry>

<entry><s:AUTHOR>กมล อดุลพันธ์</s:AUTHOR>.<s:DATE>2522</s:DATE>.<s:TITLE>การ
บริหารรัฐกิจเบื้องต้น</s:TITLE>.<s:LOCATION>กรุงเทพมหานคร</s:LOCATION>:
<s:PUBLISHER>โรงพิมพ์มหาวิทยาลัย รามคำแหง</s:PUBLISHER>.</entry>
</entries>
```

ตัวอย่างข้อมูล CORA Dataset

```
<!--<s:NAME[[:space:]]+CLASS="AUTHOR">\([^<]+\)</s:NAME>
<s:AUTHOR>\1</s:AUTHOR>-->
<entries  xmlns:s="http://example.org/state"  xmlns:sy="http://example.org/style">

<entry><s:AUTHOR>A. Cau</s:AUTHOR>, <s:AUTHOR>R. Kuiper</s:AUTHOR>, and
<s:AUTHOR>W.-P. de
Roever</s:AUTHOR>. <s:TITLE>Formalising Dijkstra's development
strategy within Stark's formalism</s:TITLE>. In <s:EDITOR>C. B. Jones</s:EDITOR>,
<s:EDITOR>R. C.
Shaw</s:EDITOR>, and <s:EDITOR>T. Denvir</s:EDITOR>, editors,
<s:BOOKTITLE>Proc. 5th. BCS-FACS Refinement Workshop</s:BOOKTITLE>,
<s:DATE>1992</s:DATE>.</entry>

<entry><s:AUTHOR>M. Kitsuregawa</s:AUTHOR>, <s:AUTHOR>H. Tanaka</s:AUTHOR>,
and <s:AUTHOR>T. Moto-oka</s:AUTHOR>. <s:TITLE>Application of hash to data
base machine and its architecture</s:TITLE>. <s:JOURNALTITLE>New
Generation Computing</s:JOURNALTITLE>,
<s:VOLUME>1</s:VOLUME>(<s:NUMBER>1</s:NUMBER>), <s:DATE>1983</s:DATE>.
</entry>

<entry><s:AUTHOR>Alexander Vrchoticky</s:AUTHOR>. <s:TITLE>Modula/R
language definition</s:TITLE>. <s:TECHTITLE>Technical Report TU Wien
rr-02-92, version 2.0</s:TECHTITLE>, <s:LOCATION>Dept. for Real-Time
Systems, Technical University of Vienna</s:LOCATION>, <s:DATE>May
1993</s:DATE>. </entry>
</entries>
```

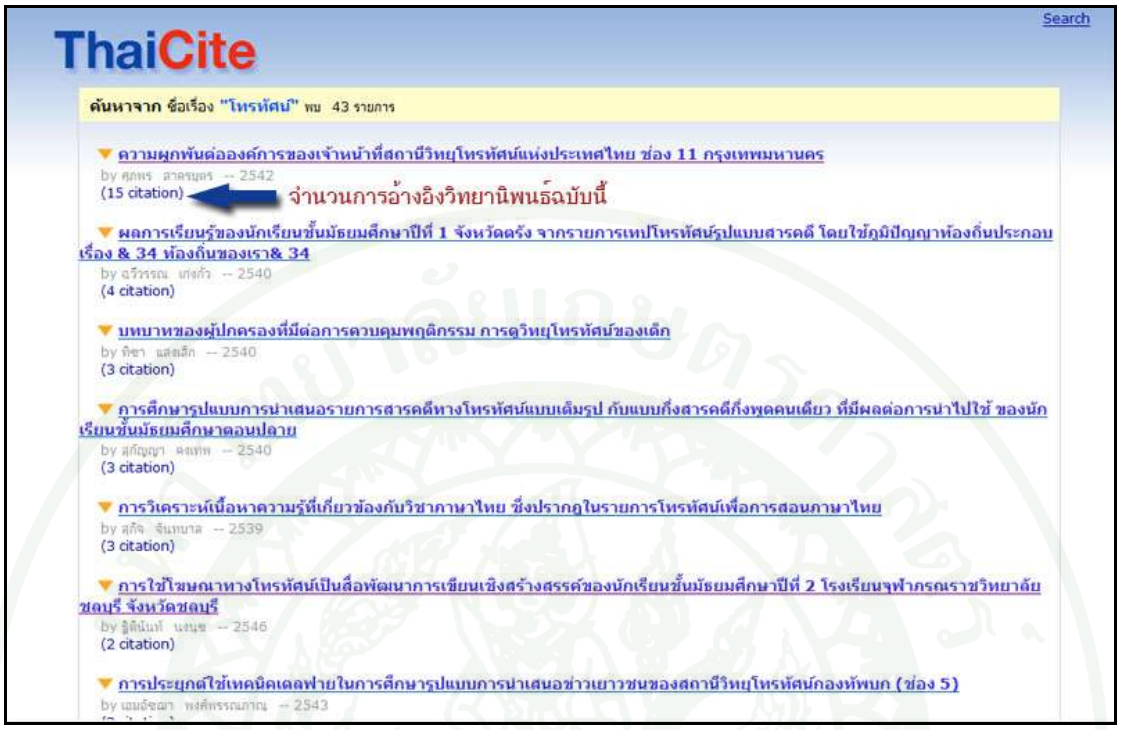
ภาคผนวก ข

คู่มือการใช้โปรแกรม ThaiCite

1. พิมพ์ URL http://ThaiCite/co.cc จะแสดง ดังภาพผนวกที่ ข1

ภาพผนวกที่ ข1 โปรแกรม ThaiCite ระบุ keyword ที่ต้องการค้นหา

2. ระบุเงื่อนไขที่ต้องการค้นหาจาก Title (ชื่อเรื่อง) หรือ Author (ผู้แต่ง)
3. ระบุ keyword ที่ต้องการค้นหา จากตัวอย่าง ระบุ keyword เป็น “โทรทัศน์” โดยให้ค้นหาจาก ชื่อ เรื่อง จากนั้นคลิกปุ่ม Search โปรแกรมจะแสดงวิทยานิพนธ์ ที่ชื่อเรื่อง มีคำว่า “โทรทัศน์” ปรากฏอยู่ ดังภาพผนวกที่ ข2



ภาพผนวกที่ ข2 วิทยานิพนธ์ที่มี keyword ตามคำที่ค้นหา และจำนวนที่วิทยานิพนธ์นั้นถูกอ้างอิง

4. ผลลัพธ์จากการค้นหา โปรแกรมจะแสดงวิทยานิพนธ์ที่ถูกอ้างอิงมากที่สุด เรียงจากมากไปน้อย ตัวอย่างเช่น วิทยานิพนธ์ที่มีคำว่า “โทรทัศน์” ปรากฏอยู่ในชื่อเรื่อง มีวิทยานิพนธ์ที่มีชื่อเรื่องว่า “ความผูกพันธ์ต่อองค์กรของเจ้าหน้าที่สถานีวิทยุโทรทัศน์แห่งประเทศไทย ช่อง 11 กรุงเทพมหานคร” ถูกอ้างอิงจำนวน 15 ฉบับ ดังภาพผนวกที่ ข2
5. หากต้องการทราบว่า วิทยานิพนธ์ฉบับดังกล่าวถูกอ้างอิงโดยวิทยานิพนธ์อะไรบ้าง ให้คลิกที่ชื่อเรื่อง โปรแกรมจะแสดงรายชื่อวิทยานิพนธ์ที่อ้างอิงวิทยานิพนธ์ดังกล่าว และแสดงกราฟแท่งจำนวนวิทยานิพนธ์ที่อ้างอิงวิทยานิพนธ์นั้นๆในแต่ละปี ดังภาพผนวกที่ ข3
6. หากต้องการดูรายละเอียดของบทความที่อ้างอิงในภาพผนวกที่ ข3 สามารถคลิกดูรายละเอียดเพิ่มเติมได้ที่ชื่อวิทยานิพนธ์

ภาพผนวกที่ ข3 จำนวนและรายชื่อวิทยานิพนธ์ที่อ้างอิงวิทยานิพนธ์ที่เลือก

ประวัติการศึกษา และการทำงาน

ชื่อ –นามสกุล	นายพิชัย กิตติคง
วัน เดือน ปี ที่เกิด	วันที่ 25 พฤศจิกายน 2523
สถานที่เกิด	ปราจีนบุรี
ประวัติการศึกษา	วศ.บ. (วิศวกรรมคอมพิวเตอร์) มหาวิทยาลัยเทคโนโลยีราชมงคลธัญบุรี
ตำแหน่งหน้าที่การงานปัจจุบัน	หัวหน้าสายงานระบบปฏิบัติการ
สถานที่ทำงานปัจจุบัน	บริษัท แพนราชเทวี กรุ๊ป จำกัด มหาชน
ผลงาน	โปสเตอร์เรื่อง การวิเคราะห์การอ้างอิงของวิทยานิพนธ์ในมหาวิทยาลัยเกษตรศาสตร์, JCSSE 2008, ระหว่างวันที่ 7-9 พฤษภาคม 2551, กาญจนบุรี งานวิจัยเรื่อง การสกัดการอ้างอิงของวิทยานิพนธ์ด้วย Context Free Grammar, NCSEC 2009, กรุงเทพ