การค้นพบโมทีฟความยาวแปรผันสำหรับข้อมูลอนุกรมเวลา

นายปวัน นันทานิช

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2554

VARIABLE LENGTH MOTIF DISCOVERY FOR TIME SERIES DATA

Mr. Pawan Nunthanid

A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Engineering Program in Computer Engineering
Department of Computer Engineering
Faculty of Engineering
Chulalongkorn University
Academic Year 2011

หัวข้อวิทยานิพนธ์	การค้นพบโมทีฟความยาวแปรผันสำหรับข้อมูลอนุกรมเวลา
โดย	นายปวัน นันทานิช
สาขาวิชา	วิศวกรรมคอมพิวเตอร์
อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก	ผู้ช่วยศาสตราจารย์ ดร.โชติรัตน์ รัตนามหัทธนะ

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญามหาบัณฑิต

………………………………………….. คณบดีคณะวิศวกรรมศาสตร์

(รองศาสตราจารย์ ดร.บุญสม เลิศหิรัญวงศ์)

คณะกรรมการสอบวิทยานิพนธ์

…………………………………………… ประธานกรรมการ

(ศาสตราจารย์ ดร.บุญเสริม กิจศิริกุล)

…………………………………………... อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก

(ผู้ช่วยศาสตราจารย์ ดร.โชติรัตน์ รัตนามหัทธนะ)

…………………………………………… กรรมการ

(ผู้ช่วยศาสตราจารย์ ดร.สุกรี สินธุภิญโญ)

…………………………………………… กรรมการภายนอกมหาวิทยาลัย

(รองศาสตราจารย์ ดร.กฤษณะ ไวยมัย)

ปวัน นันทานิช : การค้นพบโมทีฟความยาวแปรผันสำหรับข้อมูลอนุกรมเวลา. (VARIABLE LENGTH MOTIF DISCOVERY FOR TIME SERIES DATA) อ. ที่ปรึกษาวิทยานิพนธ์หลัก : ผศ. ดร.โชติรัตน์ รัตนามหัทธนะ, 61 หน้า.

การค้นพบโมทีฟของข้อมูลอนุกรมเวลาเป็นสาขาหนึ่งของงานวิจัยการทำเหมืองข้อมูลอนุกรมเวลาที่ทำหน้าที่ในการค้นหารูปแบบที่น่าสนใจที่เรียกว่าโมทีฟ โดยโมทีฟคือคู่ของลำดับย่อยในข้อมูลอนุกรมเวลาที่รูปร่างคล้ายกัน โดยทั่วไปแล้วในกระบวนการเบื้องต้นเมื่อเริ่มค้นหาโมทีฟ จะต้องกำหนดค่าของพารามิเตอร์ความยาวโมทีฟเสมอ ซึ่งงานวิจัยต่าง ๆ ไม่ได้คำนึงถึงมากนักเมื่อผู้ใช้งานต้องกำหนดความยาวโมทีฟที่จะค้นหาโดยไม่รู้แน่ชัดว่าควรกำหนดขนาดเป็นเท่าใด การกำหนดความยาวโมทีฟที่แตกต่างกันออกไปจะนำไปสู่การค้นพบรูปแบบของโมทีฟหลากหลายรูปแบบ ซึ่งมีงานจำนวนน้อยมากที่กล่าวถึงปัญหาความยาวโมทีฟและนำเสนออัลกอริทึมในการแก้ปัญหา อย่างไรก็ตามอัลกอริทึมเหล่านี้ยังต้องกำหนดค่าความยาวโมทีฟเริ่มต้นเป็นพารามิเตอร์และยังมีพารามิเตอร์อื่นเพิ่มเติมขึ้นมาอีกทำให้มีความซับซ้อนในการใช้งานรวมไปถึงยังต้องกำหนดความยาวโมทีฟเริ่มต้นอยู่ดี โดยต้องกำหนดให้มีความยาวใกล้เคียงกับรูปแบบที่น่าสนใจในข้อมูลอนุกรมเวลา ดังนั้น ปัญหาความยาวโมทีฟจึงยังคงไม่ได้รับการแก้ไข งานวิจัยในวิทยานิพนธ์ฉบับนี้จึงได้นำเสนออัลกอริทึมในการแก้ปัญหาความยาวโมทีฟซึ่งไม่ต้องการพารามิเตอร์ใด ๆ เพิ่มเติมในการใช้งานและให้ผลลัพธ์เป็นเซตของ "โมทีฟที่ดี" โดยมีวิธีการวัดคุณภาพของผลลัพธ์โมทีฟและประสิทธิภาพของอัลกอริทึมที่ชัดเจน อัลกอริทึมที่นำเสนอจะมีเพียงข้อมูลอนุกรมเวลาเป็นข้อมูลนำเข้าและได้ผลลัพธ์เป็นเซตของ “โมทีฟที่ดี” ที่ทำการจัดอันดับไว้ให้เลือกไปใช้งาน โดยอัลกอริทึมที่นำเสนอสามารถค้นพบรูปแบบที่น่าสนใจที่ทำการฝังตัวลงไปได้ทั้งหมด โดยมีคุณภาพของผลลัพธ์โมทีฟที่สูงและสามารถที่จะลดจำนวนของโมทีฟที่เป็นไปได้มากกว่า 99 เปอร์เซ็นต์

ภาควิชา วิศวกรรมคอมพิวเตอร์ ลายมือชื่อนิสิต ....

สาขาวิชา วิศวกรรมคอมพิวเตอร์ ลายมือชื่อ อ.ที่ปรึกษาวิทยานิพนธ์หลัก ....

ปีการศึกษา 2554

5370457321 : MAJOR COMPUTER ENGINEERING

KEYWORDS : MOTIF DISCOVERY / TIME SERIES / VARIABLE LENGTH / TIME SERIES MOTIF

PAWAN NUNTHANID : VARIABLE LENGTH MOTIF DISCOVERY FOR TIME SERIES DATA. ADVISOR : ASST. PROF.CHOTIRAT RATANAMAHATANA, Ph.D., 61 pp.

Time series motif discovery is an increasingly popular research area in time series mining whose main objective is to search for interesting patterns or motifs. A motif is a pair of time series subsequences, or two subsequences whose shapes are very similar to each other. Typical motif discovery algorithm requires a predefined motif length as its parameter. Discovering motif with arbitrary lengths introduces another problem, where selecting a suitable length for the motif is non-trivial since domain knowledge is often required. Only a few works were aware of this motif length and proposed some algorithms to resolve the problem. However, these algorithms still require an initial motif length parameter and many additional pre-defined parameters which cause a lot more complication for using and especially the motif length parameter is still remain. Thus, this work proposes the first parameter-free motif discovery algorithm which requires no parameter as input, and as a result returns a set of all “Best Motif” that are ranked by a proposed scoring function which is based on similarity of motif locations and similarity of motif shapes. The experimental results show that the algorithm can efficiently discover all planted patterns with high quality and are able to reduce a number of all possible motifs with more than 99 percent.

Department : Computer Engineering	Student's Signature
Field of Study : Computer Engineering	Advisor's Signature
Academic Year : 2011	

กิตติกรรมประกาศ

งานวิจัยในวิทยานิพนธ์ฉบับนี้จะไม่มีทางสำเร็จลุล่วงไปได้ด้วยความสามารถของข้าพเจ้าเพียงคนเดียว หากไม่ได้รับการสนับสนุนจากผู้มีพระคุณหลาย ๆ ท่านที่กรุณาช่วยเหลือข้าพเจ้าตลอดมาในช่วงระยะเวลาการศึกษานี้ แม้เต็มไปด้วยความยากลำบากและอุปสรรคนานัปการแต่ก็เป็นประสบการณ์อันมีค่ายิ่งที่สั่งสอนบ่มเพาะฝึกฝนข้าพเจ้า ให้รู้จักการเรียนรู้ด้วยตนเอง การคิดอย่างเป็นเหตุเป็นผล ให้มีความอดทน รับผิดชอบและขยันหมั่นเพียร

ขอขอบพระคุณอาจารย์ที่ปรึกษาวิทยานิพนธ์ฉบับนี้ ผู้ช่วยศาสตราจารย์ ดร.โชติรัตน์ รัตนามหัทธนะ ที่กรุณาสละเวลาอันมีค่าช่วยชี้แนะแนวทางในการทำวิจัยและอบรมแก้ไขศิษย์คนนี้ด้วยดีเสมอมา

ขอขอบพระคุณ ดร.วิชญ์ เนียรนาทตระกูล ซึ่งเป็นทั้งเพื่อนและรุ่นพี่ที่ห้องปฏิบัติการที่คอยช่วยสนับสนุน ชี้แนะแนวทางและให้ข้อคิดต่าง ๆ ในการทำวิจัยตลอดมา

ขอขอบพระคุณคณะกรรมการสอบวิทยานิพนธ์ฉบับนี้ ซึ่งประกอบไปด้วยศาสตราจารย์ ดร.บุญเสริม กิจศิริกุล ผู้ช่วยศาสตราจารย์ ดร.สุกรี สินธุภิญโญ และรองศาสตราจารย์ ดร.กฤษณะ ไวยมัย ที่ให้แนวทางแก้ไขและข้อคิดในงานวิจัยที่ยังมีจุดบกพร่องเพื่อพัฒนาแก้ไขให้ดียิ่ง ๆ ขึ้นไป

ขอขอบคุณเพื่อน ๆ ทั้งในและนอกห้องปฏิบัติการทุกคนที่คอยช่วยเหลือซึ่งกันและกันในทุก ๆ ด้านและร่วมทุกข์ร่วมสุขด้วยกันมาตลอดระยะเวลาการศึกษา

สุดท้ายนี้ ขอขอบพระคุณบิดามารดาและน้องสาวของข้าพเจ้าที่เข้าใจและเป็นกำลังใจให้ในช่วงเวลาที่ยากลำบากยิ่งเสมอมา

สารบัญ

หน้า

หน้า

หน้า

สารบัญตาราง

หน้า

สารบัญภาพ

หน้า

หน้า

หน้า

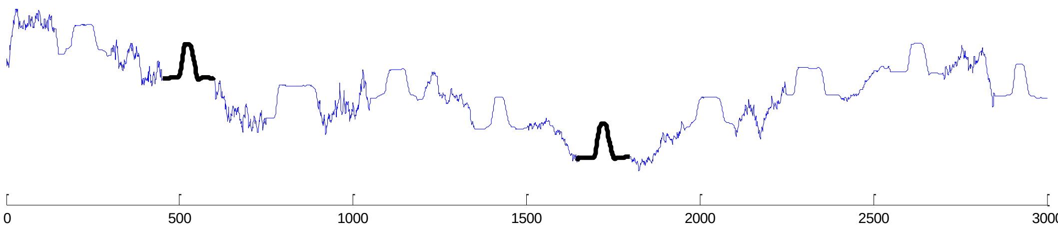
หน้า

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

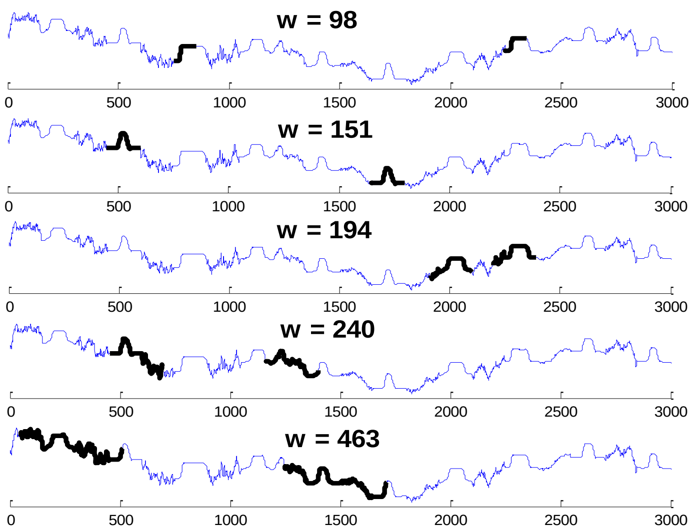
เนื่องด้วยข้อมูลจำนวนมหาศาลเกิดขึ้นอย่างรวดเร็วในแต่ละวัน การนำข้อมูลเหล่านี้มาใช้งานจึงเป็นไปได้ยาก ดังนั้น เทคนิคของการทำเหมืองข้อมูลจำนวนมากจึงได้รับการพัฒนาขึ้นเพื่อตอบสนองความต้องการของผู้ใช้งานข้อมูลแต่ละประเภท ข้อมูลอนุกรมเวลาเป็นข้อมูลที่มีการใช้งานอย่างแพร่หลายชนิดหนึ่งในแอพพลิเคชันจำนวนมากและมีเทคนิคหนึ่งที่เป็นที่รู้จักดีในข้อมูลชนิดนี้ เรียกว่า การค้นพบโมทีฟในข้อมูลอนุกรมเวลา เทคนิคนี้จะเน้นที่การค้นหารูปแบบคล้าย ๆ กันที่เกิดขึ้นในลำดับของข้อมูลอนุกรมเวลาที่เรียกว่า “โมทีฟ” วิธีการหาโมทีฟนี้ค่าของความยาวโมทีฟต้องได้รับการกำหนดก่อน จากนั้นวิธีการที่ง่ายที่สุดคือการวัดระยะทางระหว่างลำดับย่อยทั้งหมดที่สกัดออกมาจากข้อมูลอนุกรมเวลาที่ความยาวที่กำหนด คู่ของลำดับย่อยที่มีระยะทางน้อยที่สุดจะเป็นโมทีฟของข้อมูลอนุกรมเวลานั้น ดังแสดงในภาพที่ 1.1



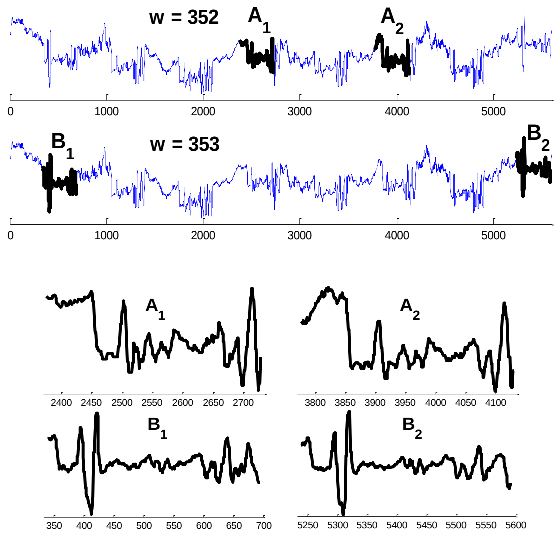
ภาพที่ 1.1 โมทีฟของข้อมูล Gun-Point [1] ขนาด 150 จุดข้อมูลถูกค้นพบที่ตำแหน่ง 450 ถึง 599 และ 1,647 ถึง 1,796 ในข้อมูลอนุกรมเวลาขนาด 3,000 จุดข้อมูล

ดังที่ได้กล่าวมาแล้วในเบื้องต้นว่าอัลกอริทึมในการหาโมทีฟในปัจจุบันยังต้องการการกำหนดค่าความยาวของโมทีฟเป็นพารามิเตอร์ เพื่อให้เห็นความสำคัญของพารามิเตอร์ความยาว ในงานวิจัยชิ้นนี้จึงทำการสาธิตผลลัพธ์โมทีฟที่ได้จากการเปลี่ยนแปลงค่าความยาวและพบว่าความยาวโมทีฟที่กำหนดให้แตกต่างกันส่งผลให้ผลลัพธ์โมทีฟแตกต่างกันไปด้วย ดังแสดงในภาพที่ 1.2 และความแตกต่างเพียงเล็กน้อยของความยาวโมทีฟอาจส่งผลให้โมทีฟที่ค้นพบนั้นแตกต่างกันโดยสิ้นเชิง ดังแสดงในภาพที่ 1.3 จากผลลัพธ์ที่ได้ ซึ่งสามารถรู้ได้ยากว่าควรเลือกโมทีฟที่ความยาวใด แสดงให้เห็นว่าการเลือกความยาวโมทีฟนั้นแฝงไว้ด้วยความยากในการเลือกโดยเฉพาะอย่างยิ่งกับผู้ใช้ที่ไม่มีความเชี่ยวชาญในข้อมูลด้านนั้น ๆ ปัญหานี้จึงมีความท้าทายอย่างยิ่งเพราะเป็นการยากที่จะรู้อย่างแน่ชัดว่าโมทีฟที่ความยาวเท่าใดจึงจะเหมาะสม ซึ่งยังไม่ได้รวมไปถึงความต้องการของการประมวลผลที่เพิ่มมากขึ้นสำหรับปัญหา ปกติแล้วความยาวโมทีฟสามารถกำหนด

ได้ในหลาย ๆ ทาง เช่น โดยผู้เชี่ยวชาญ โดยการลองผิดลองถูก โดยการสังเกตลักษณะของข้อมูลอนุกรมเวลาหรือบางครั้งโดยการทดลอง อย่างไรก็ตามไม่มีทางใดที่รับรองได้ว่าความยาวโมทีฟที่เลือกนั้นจะมีความเหมาะสมที่สุด แม้แต่โดยการเลือกจากผู้เชี่ยวชาญ ดังนั้น ตัวเลือกของพารามิเตอร์นี้จึงมีความสำคัญอย่างยิ่ง



ภาพที่ 1.2 ผลลัพธ์ของโมทีฟที่ถูกค้นพบในข้อมูล Gun-Point [1] เมื่อกำหนดขนาดความยาว (*w*) แตกต่างกันจำนวน 5 โมทีฟ ซึ่งรูปร่างและตำแหน่งของโมทีฟแตกต่างกันอย่างชัดเจน



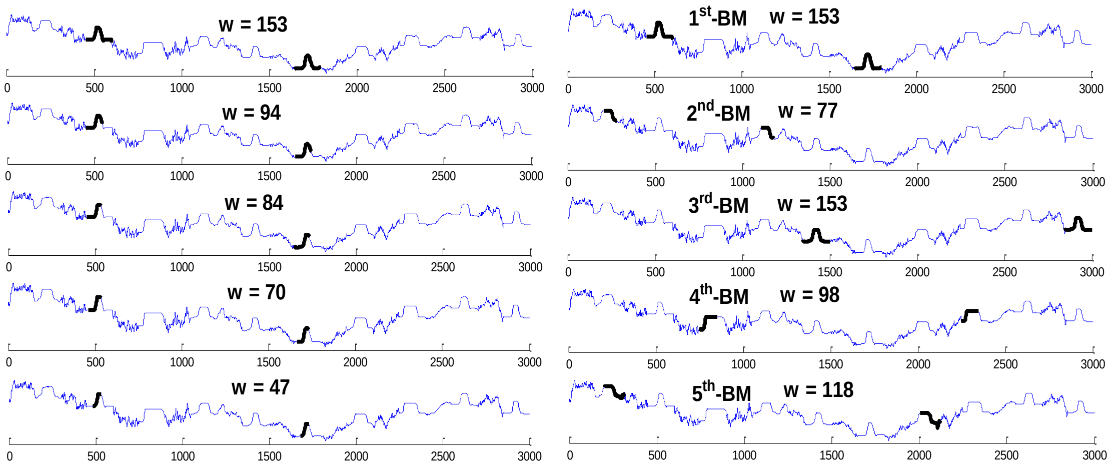
ภาพที่ 1.3 โมทีฟของข้อมูล Face-Four [1] ขนาดความยาว 352 และ 353 จุดข้อมูล ตามลำดับ คู่แรก A_1 และ A_2 พบที่ตำแหน่ง 2,376 และ 3,777 ส่วนคู่ที่สอง B_1 และ B_2 พบที่ตำแหน่ง 341 และ 5,239

แม้ว่าจะมีงานวิจัยที่เกี่ยวกับการค้นพบโมทีฟจำนวนหลายงานวิจัย [2][3][4][5][6][7][8][9] ซึ่งโดยส่วนใหญ่แล้วเน้นที่การเพิ่มความเร็วของการหาโมทีฟเท่านั้น มีงานอยู่จำนวนน้อยมากที่กล่าวถึงปัญหาการกำหนดความยาวของโมทีฟ จากการค้นคว้ามีงานวิจัยจำนวนทั้งสิ้นเพียง 2 งานวิจัย คือ

1. Tang และ Liao [10] ได้นำเสนออัลกอริทึมที่เรียกว่า การต่อจุดโมทีฟ (Motif Concatenation)

2. Li และ Lin [11] ได้นำเสนออัลกอริทึมที่มีพื้นฐานมาจากการเหนี่ยวนำไวยากรณ์ (Grammar Inference)

อย่างไรก็ตาม ทั้ง 2 งานวิจัยนี้ต้องการพารามิเตอร์อื่น ๆ เพิ่มเติมโดยเฉพาะอย่างยิ่งค่าของความยาวโมทีฟเริ่มต้นซึ่งต้องกำหนดให้ใกล้เคียงกับรูปแบบที่น่าสนใจให้มากที่สุดเพื่อเพิ่มโอกาสในการค้นพบรูปแบบที่น่าสนใจ ปัญหาของความยาวโมทีฟจึงยังไม่ได้รับการแก้ไข เนื่องจากในอัลกอริทึมยังต้องกำหนดค่าความยาวโมทีฟเหมือนเดิม และด้วยพารามิเตอร์ที่เพิ่มขึ้นมาเหล่านี้ทำให้อัลกอริทึมในการหาโมทีฟยังไม่เหมาะกับการนำไปใช้งานจริง ดังนั้น ปัญหาความยาวของโมทีฟจึงยังคงอยู่ ดังนั้น งานวิจัยนี้ จึงเป็นงานวิจัยแรกที่แก้ไขปัญหาของความยาวโมทีฟ โดยนำเสนออัลกอริทึมที่ไม่มีพารามิเตอร์เป็นอัลกอริทึมแรก เรียกว่า *k*-Best Motif Discovery (*k*BMD) เพื่อแก้ปัญหาความยาวของโมทีฟ โดยอัลกอริทึมนี้ พัฒนาจากแนวคิดหลักของ “โมทีฟที่ดี” ซึ่งหมายถึง โมทีฟที่เป็นตัวแทนของกลุ่มโมทีฟที่มีความยาวแตกต่างกันและเกิดขึ้นในตำแหน่งที่ใกล้เคียงกัน โดยแนวคิดนี้แสดงไว้ดังภาพที่ 1.4 ก) โดย *k*BMD ต้องการเพียงข้อมูลอนุกรมเวลาเป็นข้อมูลนำเข้าเท่านั้นและให้ผลลัพธ์เป็นเซตของ “โมทีฟที่ดี” ซึ่งจัดอันดับไว้ดังตัวอย่างในภาพที่ 1.4 ข)



ก) ข)

ภาพที่ 1.4 แนวคิดของ “โมทีฟที่ดี” โดยภาพ ก) แสดงโมทีฟที่ความยาว 153, 94, 84, 70, 47 จุดข้อมูล ที่ถูกค้นพบที่ตำแหน่งใกล้เคียงกัน เนื่องจากตำแหน่งเหล่านี้มีรูปร่างคล้ายกัน จำนวนโมทีฟที่ความยาวต่างกันจึงถูกค้นพบที่ตำแหน่งนี้จำนวนมาก ดังนั้น โมทีฟที่ยาวที่สุด (w=153) จึงถูกเลือกให้เป็นตัวแทน ซึ่งเป็น “โมทีฟที่ดี” ของกลุ่มโมทีฟกลุ่มนี้ ส่วนในภาพ ข) แสดงผลลัพธ์ “โมทีฟที่ดี” (Best Motif - BM) จำนวน 5 โมทีฟที่ถูกจัดอันดับโดยอัลกอริทึมการให้คะแนนที่นำเสนอ

1.2 วัตถุประสงค์ของการวิจัย

งานวิจัยนี้มีวัตถุประสงค์เพื่อนำเสนอวิธีการหาโมทีฟของข้อมูลอนุกรมเวลาโดยไม่ต้องกำหนดพารามิเตอร์ความยาวโมทีฟ

1.3 ขอบเขตของการวิจัย

1. งานวิจัยนี้ทดลองกับข้อมูลอนุกรมเวลาแบบจำแนกประเภทจาก UCR [1]

2. อัลกอริทึมที่นำเสนอมีข้อมูลขาเข้า คือ ข้อมูลอนุกรมเวลา และผลลัพธ์ที่เป็นข้อมูลออก คือ เซตของโมทีฟ

3. อัลกอริทึมที่นำเสนอวิธีการกำหนดความยาวโมทีฟ สามารถลดจำนวนคำตอบของความยาวโมทีฟให้น้อยลงโดยไม่ต้องกำหนดค่าความยาวในการหาโมทีฟ

4. ประเมินผลเปรียบเทียบกับอัลกอริทึมของงานวิจัยที่ผ่านมา

1.4 ข้อจำกัดของการวิจัย

เนื่องจากงานวิจัยชิ้นนี้เป็นงานวิจัยแรก ๆ ของการหาโมทีฟที่ความยาวที่เหมาะสม ที่เน้นถึงความถูกต้องของผลลัพธ์โมทีฟเป็นหลักก่อน จึงมีขั้นตอนที่ต้องค้นหาโมทีฟที่ทุกความยาวที่เป็นไปได้ออกมาเพื่อทำการวิเคราะห์ ด้วยขั้นตอนนี้ทำให้ต้องใช้เวลาในการประมวลผลค่อนข้างมากซึ่งจะเป็นปัญหาที่จะทำการแก้ไขต่อไปในอนาคต

1.5 ประโยชน์ที่จะได้รับ

สามารถหาโมทีฟของข้อมูลอนุกรมเวลาได้โดยไม่ต้องกำหนดพารามิเตอร์ความยาวโมทีฟด้วยวิธีการที่นำเสนอ

1.6 วิธีดำเนินการวิจัย

1. ศึกษางานวิจัยที่ผ่านมาที่เกี่ยวข้องกับโมทีฟของข้อมูลอนุกรมเวลา

2. ออกแบบวิธีการวัดผลการทดลองที่สามารถวัดผลกับอัลกอริทึมสำหรับค้นหาโมทีฟโดยไม่ต้องกำหนดพารามิเตอร์ความยาว

3. เตรียมข้อมูลเพื่อใช้ในการทดลอง

4. ออกแบบและพัฒนาอัลกอริทึม

5. นำข้อมูลที่สร้างไว้มาทำการทดลองกับอัลกอริทึมที่พัฒนาแล้วบันทึกผล

6. ทำซ้ำอัลกอริทึมของงานวิจัยที่เกี่ยวข้องเพื่อวัดผลเปรียบเทียบ

7. วิเคราะห์และสรุปผลการทดลอง

8. จัดทำวิทยานิพนธ์

1.7 ลำดับขั้นตอนในการเสนอผลการวิจัย

ส่วนหนึ่งของงานวิทยานิพนธ์นี้ ได้รับการตีพิมพ์เป็นบทความทางวิชาการจำนวน 2 เรื่อง ดังนี้

1. “Discovery of Variable Length Time Series Motif” โดย ปวัน นันทานิช วิชญ์ เนียรนาทตระกูล และ โชติรัตน์ รัตนามหัทธนะ ในงานประชุมวิชาการ “The Eighth Annual International Conference Organized by Electrical Engineering/ Electronics, Computer, Telecommunications and Information Technology (ECTI) Association, Thailand” ซึ่งจัดขึ้น ณ จังหวัดขอนแก่น ประเทศไทย ระหว่างวันที่ 17 พฤษภาคม ถึง 19 พฤษภาคม 2554

2. “Parameter Free Motif Discovery for Time Series Data” โดย ปวัน นันทานิช วิชญ์ เนียรนาทตระกูล และ โชติรัตน์ รัตนามหัทธนะ ในงานประชุมวิชาการ “The ninth Annual International Conference Organized by Electrical Engineering/ Electronics, Computer, Telecommunications and Information Technology (ECTI) Association, Thailand” ซึ่งจัดขึ้น ณ จังหวัดประจวบคีรีขันธ์ ประเทศไทย ระหว่างวันที่ 16 พฤษภาคม ถึง 18 พฤษภาคม 2555

บทที่ 2
งานวิจัยที่เกี่ยวข้อง

ในหัวข้อทฤษฎีที่เกี่ยวข้องนี้จะกล่าวถึงหลักการของแต่ละทฤษฎีที่ได้รับการนำไปใช้ในขั้นตอนของอัลกอริทึมที่นำเสนอ จากนั้นในส่วนของงานวิจัยที่เกี่ยวข้องจะแสดงถึงวิธีการและขั้นตอนต่าง ๆ ที่เกี่ยวกับการหาโมทีฟของข้อมูลอนุกรมเวลารวมไปถึงงานวิจัยที่นำเสนอวิธีการค้นพบโมทีฟความยาวแปรผันของข้อมูลอนุกรมเวลาพร้อมทั้งการวิเคราะห์และวิจารณ์งานที่ผ่านมาเหล่านั้น

2.1 แนวคิดและทฤษฎี

เมื่อกล่าวถึงการค้นพบโมทีฟของข้อมูลอนุกรมเวลาซึ่งต้องวัดความคล้ายกันของแต่ละลำดับย่อยเพื่อหาคู่ที่ใกล้เคียงกันมากที่สุดจำเป็นต้องมีมาตรวัดความคล้ายซึ่งในอัลกอริทึมที่นำเสนอนี้จะใช้ระยะทางยุคลิดเป็นตัววัดความคล้ายของลำดับย่อยโดยระยะทางยุคลิด จะมีคุณสมบัติของมาตรวัดระยะทาง (Distance Metric) ซึ่งเป็นหลักการหนึ่งในทฤษฎีที่เกี่ยวข้อง จากนั้นต่อเนื่องจากคุณสมบัติหนึ่งของตัววัดระยะทางนี้ คือ คุณสมบัติความไม่เท่ากันของสามเหลี่ยมซึ่งเป็นอีกหนึ่งทฤษฎีที่สำคัญที่ได้รับการนำไปใช้ในอัลกอริทึมการค้นหาโมทีฟแบบ MK (Mueen-Keogh) [9] ของงานวิจัยที่เกี่ยวข้อง โดยในอัลกอริทึมที่นำเสนอนี้ได้นำ MK มาใช้ในขั้นตอนการหาโมทีฟ จึงนำมากล่าวไว้ในหัวข้อนี้เป็นเบื้องต้น

2.1.1 มาตรวัดระยะทาง (Distance Metric)

มาตรวัดระยะทางหรือฟังก์ชันที่เกี่ยวกับการวัดระยะทางที่เป็นตัวกำหนดระยะระหว่างวัตถุต่าง ๆ ที่อยู่ในเซต ซึ่งไม่ใช่ทุกเซตที่มีโครงสร้างเป็นมาตรวัดระยะทาง จะมีเฉพาะบางโครงสร้างเท่านั้นที่สามารถอธิบายได้โดยใช้คุณสมบัติ 4 ข้อของมาตรวัดระยะทาง ซึ่งมีดังต่อไปนี้

เซต X บน Metric เป็นฟังก์ชันระยะทาง $d : X \times X \longrightarrow R$ โดย R เป็นเซตของจำนวนจริงและมีวัตถุ x, y, z อยู่ใน X จะมีคุณสมบัติดังนี้

1. $d(x, y) \geq 0$ ระยะทางจาก x ไป y จะไม่มีค่าติดลบ (Non-Negativity)

2. $d(x, y) = 0$ ระยะทางจาก x ไป y จะเป็น 0 ถ้า $x = y$ (Identity of Indiscernible)

3. d(x, y) = d(y, x) ระยะทางจาก x ไป y จะเท่ากับ y ไป x (Symmetry)

4. d(x, z) ≤ d(x, y) + d(y, z) อสมการความไม่เท่ากันของสามเหลี่ยม (Triangular Inequality)

2.1.2 **ระยะทางยุคลิด** (Euclidean Distance)

ระยะทางยุคลิดเป็นตัววัดระยะทางรูปแบบหนึ่งที่มีคุณสมบัติของมาตรวัดระยะทางทั้ง 4 ข้อตามที่กล่าวมาข้างต้น ซึ่งเป็นวิธีการวัดระยะทางระหว่างจุดสองจุดโดยมีคำจำกัดความดังต่อไปนี้

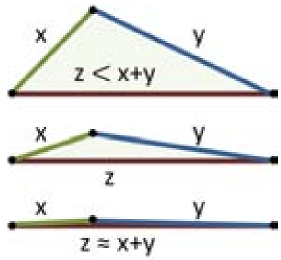
ระยะทางยุคลิดระหว่างจุด p และ q เป็นความยาวของเส้นเชื่อมต่อระหว่างจุดกำหนดให้ $p = (p_1, p_2,..., p_n)$ และ $q = (q_1, q_2,..., q_n)$ เป็นสองจุดที่อยู่บนพื้นที่ยุคลิดแล้วระยะทางระหว่าง p ถึง q คือ

$$\mathrm{d}(p,q)=\mathrm{d}(q,p)=\sqrt{(q_1-p_1)^2+(q_2-p_2)^2+...+(q_n-p_n)^2}=\sqrt{\sum_{i=1}^{n}(q_i-p_i)^2} \qquad (2.1)$$

2.1.3 **คุณสมบัติความไม่เท่ากันของสามเหลี่ยม** (Triangular Inequality)

ทฤษฎีบทความไม่เท่ากันของสามเหลี่ยม เป็นทฤษฎีหนึ่งที่ใช้กันอย่างกว้างขวางในงานวิจัยด้านการค้นพบโมทีฟของข้อมูลอนุกรมเวลา เนื่องจากทฤษฎีความไม่เท่ากันของสามเหลี่ยมนี้บรรจุอยู่ในคุณสมบัติหนึ่งจากทั้งหมดสี่ข้อของมาตรวัดระยะทาง (Distance Metric) ซึ่งสามารถใช้เป็นค่าขอบเขตล่างเพื่อลดจำนวนการคำนวณหาระยะทางยุคลิดของข้อมูลอนุกรมเวลาได้ โดยมีคุณสมบัติดังนี้

$x + y > z$ ความยาวด้าน x รวมกับความยาวด้าน y จะมากกว่าหรือเท่ากับด้าน z เสมอ ดังภาพที่ 2.1 ซึ่งถ้าเรามอง 3 จุดในรูปให้เป็นจุดที่อยู่บนพื้นที่ยุคลิดจะสังเกตได้ว่าถ้าเรารู้ระยะทางของด้าน x และ z เราจะสามารถประมาณค่าของด้าน y ได้ว่า $|x-z| \leq y$ ซึ่งจะนำค่านี้ไปใช้เป็นค่าของขอบเขตล่าง เพื่อลดจำนวนของการคำนวณที่ไม่จำเป็นออกไป



ภาพที่ 2.1 อสมการความไม่เท่ากันของสามเหลี่ยม

(ที่มา : http://en.wikipedia.org/wiki/Triangle_inequality)

2.2 งานวิจัยที่เกี่ยวข้อง

ในหัวข้อนี้จะกล่าวถึงงานวิจัยที่ผ่านมาที่ได้นำเสนออัลกอริทึมที่ใช้ในการค้นพบโมทีฟทั้งในส่วนของขั้นตอนและวิธีการ รวมไปถึงงานวิจัยที่ได้นำเสนออัลกอริทึมในการแก้ปัญหาความยาวโมทีฟซึ่งเป็นปัญหาหลักของงานวิจัยในวิทยานิพนธ์ฉบับนี้

2.2.1 การหาโมทีฟในข้อมูลอนุกรมเวลา [12]

งานวิจัยชิ้นนี้นำเสนอวิธีการค้นพบโมทีฟโดยประมาณของข้อมูลอนุกรมเวลาโดยเริ่มต้นด้วยการลดมิติข้อมูลก่อนนำไปค้นหาโมทีฟและเสนอคำจำกัดความใหม่เนื่องจากมีความคลุมเครือในการเปรียบเทียบความใกล้เคียงของข้อมูลอนุกรมเวลาเมื่อตำแหน่งของคู่เปรียบเทียบต่างเพียงเล็กน้อยซึ่งกรณีนี้จะทำให้การวัดระยะของคู่เปรียบเทียบมีความคล้ายคลึงกันมากซึ่งเป็นสิ่งที่ไม่ควรจะเกิดขึ้นเพราะจะทำให้โมทีฟที่ได้อยู่ที่ตำแหน่งใกล้เคียงและเกยทับเหมือนเป็นตัวเดียวกัน

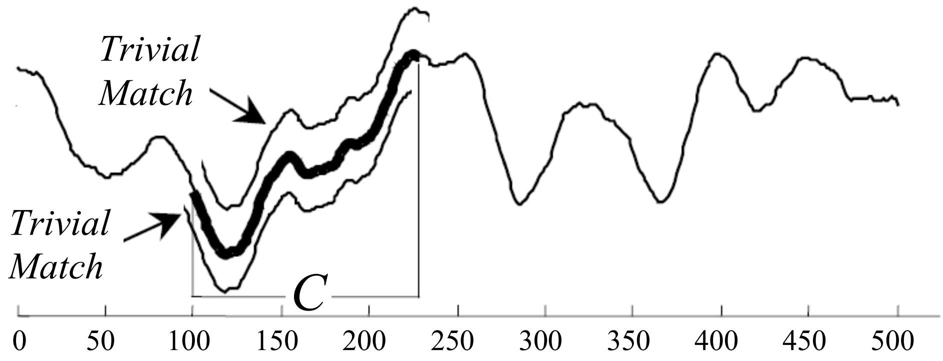
คำจำกัดความที่สำคัญต่าง ๆ ถูกกำหนดขึ้นเพื่อขจัดความคลุมเครือของการจัดลำดับย่อยให้อยู่ในชุดของโมทีฟของข้อมูลอนุกรมเวลา มีดังต่อไปนี้

1. Time Series : ข้อมูลอนุกรมเวลา $T = t_1,\ldots,t_m$ คือ ชุดลำดับของจำนวนจริงจำนวน m ตัว

2. Subsequence (ลำดับย่อย) : ถ้ามีลำดับของข้อมูลอนุกรมเวลา T ที่ขนาดความยาว m แล้วลำดับย่อย C ใน T ที่ขนาดความยาว $n < m$ คือ $C = t_p,\ldots,t_{p+n-1}$ เมื่อ $1 \le p \le m-n+1$

3. Match : เมื่อตัวแปร R เป็นระยะพิสัยที่เป็นจำนวนจริงบวกที่ถูกกำหนดขึ้น และข้อมูลอนุกรมเวลา T มีลำดับย่อย C ที่เริ่มต้นที่ตำแหน่ง p และลำดับย่อย M เริ่มต้นที่ตำแหน่ง q แล้ว C และ M จะมีความสัมพันธ์ตามคำจำกัดความ Match ก็ต่อเมื่อ $D(C, M) \leq R$ โดยที่ $D(C, M)$ คือ ระยะทางยุคลิดระหว่างลำดับย่อย C และ M

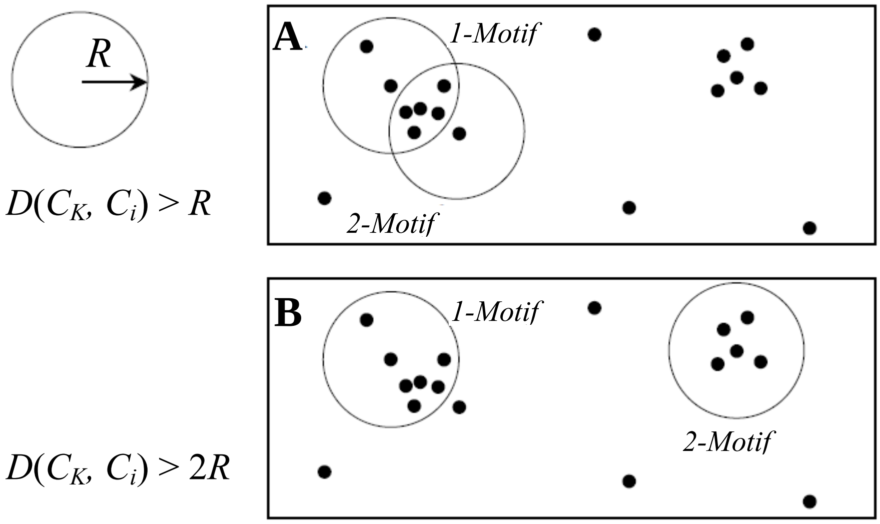
4. Trivial Match : ถ้ามีข้อมูลอนุกรมเวลา T ซึ่งมีลำดับย่อย C ที่ตำแหน่ง p และมีลำดับย่อย M ที่ตำแหน่ง q ที่มีความสัมพันธ์แบบ Match กับลำดับย่อย C แล้ว เราจะเรียก M กับ C ว่ามีความสัมพันธ์แบบ Trivial Match ก็ต่อเมื่อ $p=q$ หรือ ไม่ปรากฏลำดับย่อย M' ใด ๆ อีกแล้วที่ $D(C, M') > R$ โดยที่ $q < q' < p$ หรือ $p < q' < q$ ดังแสดงไว้ในภาพที่ 2.2



ภาพที่ 2.2 ลำดับย่อย C ในข้อมูลอนุกรมเวลาจะมีลักษณะคล้ายกับลำดับย่อยที่อยู่ในตำแหน่งถัดไปทางซ้ายและขวาที่ติดกับ C

(ที่มา : Lin, J., Keogh, E., Lonardi, S. and Patel, P. Finding Motifs in Time Series. The Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining 2002)

5. K-Motifs : ถ้ามีข้อมูลอนุกรมเวลา T และลำดับย่อยความยาว n และมีระยะพิสัย R จะได้โมทีฟของข้อมูลอนุกรมเวลาที่สำคัญที่สุดที่เรียกว่า 1-Motif คือ ลำดับย่อย C_1 ที่มีลำดับย่อยอื่น ๆ มามีความสัมพันธ์แบบ Non-Trivial Match กับตัวมันเองจำนวนมากที่สุด ดังนั้น K-Motifs จึงหมายถึงโมทีฟของข้อมูลอนุกรมเวลาที่มีความสำคัญเป็นอันดับที่ K ซึ่งก็คือลำดับย่อย C_k ที่มีจำนวนลำดับย่อยอื่น ๆ มามีความสัมพันธ์แบบ Non-Trivial Match กับตัวมันเองจำนวนมากเป็นอันดับที่ K โดยมีระยะทางยุคลิดระหว่างลำดับย่อยแต่ละตัวที่เป็นโมทีฟในแต่ละอันดับ $D(C_K, C_i) > 2R$ โดยที่ $1 \leq i < K$ ซึ่งมีแนวคิดแสดงไว้ดังภาพที่ 2.3



ภาพที่ 2.3 การอธิบายคำจำกัดความของ *K*-Motifs ด้วยรูปภาพว่าเหตุใดระยะทางยุคลิดของโมทีฟแต่ละโมทีฟถึงต้องมีระยะห่างจากกันมากกว่า 2*R* ขึ้นไป ซึ่งมีเหตุผลคือถ้าใช้เพียงระยะทาง R จะทำให้ลำดับย่อยที่เป็นโมทีฟแต่ละตัวใน *K*-Motifs เกิดการใช้ลำดับย่อยอื่น ๆ ร่วมกันดังรูป A ซึ่งไม่ควรจะเป็น ในทางตรงข้ามถ้าระยะทางมากกว่า 2*R* จะได้ โมทีฟที่ไม่เกิดการใช้ลำดับย่อยอื่นร่วมกันซึ่งจะได้โมทีฟที่เฉพาะตัวมากกว่า

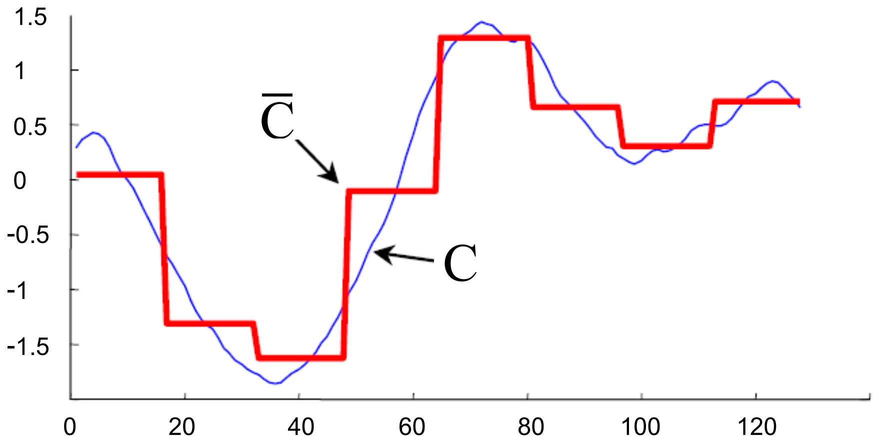
(ที่มา : Lin, J., Keogh, E., Lonardi, S. and Patel, P. Finding Motifs in Time Series. The Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining 2002)

หลังจากกำหนดคำจำกัดความแล้วจึงได้นำเสนออัลกอริทึมในการหาโมทีฟของข้อมูลอนุกรมเวลา โดยวิธีการของอัลกอริทึมนี้จะเริ่มต้นด้วยการลดมิติของข้อมูลอนุกรมเวลาโดยใช้ Piecewise Aggregate Approximation (PAA) [12] ซึ่งเป็นการทำให้ข้อมูลอนุกรมเวลาซึ่งเป็นข้อมูลต่อเนื่องเปลี่ยนเป็นข้อมูลวิยุตซึ่งแทนที่ด้วยข้อมูลที่เป็นตัวอักษร เมื่อลำดับย่อยแต่ละลำดับผ่านกระบวนการลดมิติข้อมูลแล้วจึงนำมาเปรียบเทียบกันเพื่อหาระยะทางของแต่ละลำดับ

วิธีการของ PAA ประกอบด้วย 3 ขั้นตอน ถ้าเรามีข้อมูลอนุกรมเวลา C ขนาดความยาว n อยู่ในรูป $C = C_1, \ldots, C_n$ ขั้นตอนแรกจะเริ่มด้วยการลดจำนวนมิติของข้อมูลจำนวน n มิติ ให้เหลือจำนวน w มิติ ในรูป $\overline{C} = \overline{C_1}, \ldots, \overline{C_w}$ โดยสามารถคำนวณค่าของ $\overline{C_i}$ แต่ละตัวด้วยสมการ

$$\overline{C_i} = \frac{w}{n} \sum_{j=\frac{n}{w}(i-1)+1}^{\frac{n}{w}i} C_j \qquad (2.2)$$

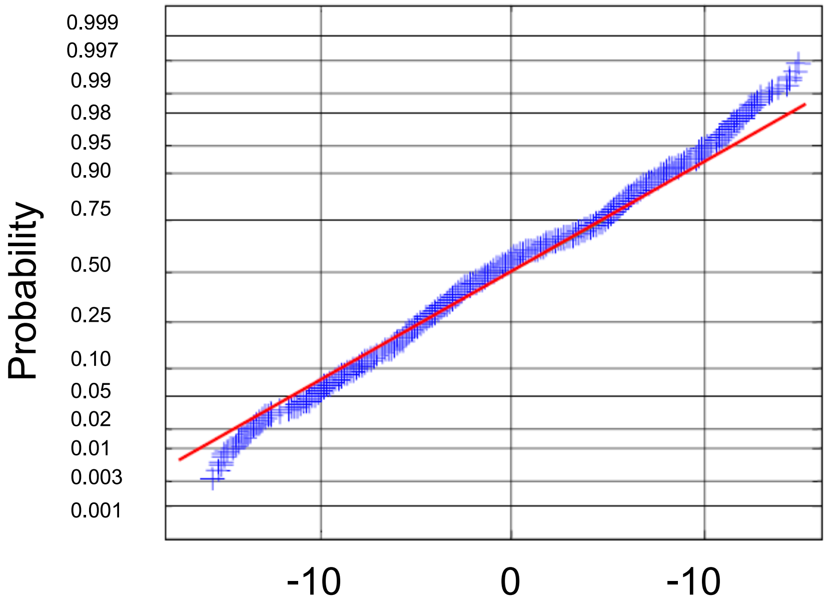
ซึ่งจะได้ค่าของข้อมูลอนุกรมเวลาออกมาเป็น ดังภาพที่ 2.4 ซึ่งในที่นี้ *C* มีขนาด *n* คือ 128 มิติ ลดลงให้เหลือเท่ากับ *w* คือ 8 มิติ จากนั้นขั้นตอนที่สองของวิธีการ PAA คือการทำให้ข้อมูลอนุกรมเวลา $\bar{C}$ ที่เหลือขนาด 8 มิตินี้ให้เป็นข้อมูลวิยุตโดยการเปลี่ยนเป็นตัวอักษร เนื่องจากข้อมูลอนุกรมเวลาที่ผ่านการทำให้อยู่ในรูปปกติแล้วจะมีการกระจายของข้อมูลแบบเกาส์เซียนซึ่งทดสอบได้ด้วยการ นำลำดับย่อยของข้อมูลอนุกรมเวลาที่แตกต่างกันจำนวน 8 ชุดข้อมูลมาสร้างกราฟความน่าจะเป็นแบบปกติของแต่ละชุดข้อมูลซึ่งผลที่ได้เป็นดังภาพที่ 2.5 ซึ่งแสดงถึงข้อมูลที่เป็นลักษณะของสมการเส้นตรงที่เป็นลักษณะของการกระจายของข้อมูลแบบเกาส์เซียน



ภาพที่ 2.4 แสดงการนำเสนอการลดมิติข้อมูลของวิธีการ PAA ซึ่งลดมิติจากขนาดเท่าความยาวของข้อมูลอนุกรมเวลา *C* คือ 128 เหลือ 8 มิติ

(ที่มา : Lin, J., Keogh, E., Lonardi, S. and Patel, P. Finding Motifs in Time Series. The Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining 2002)

เมื่อข้อมูลอนุกรมเวลาที่ถูกทำให้อยู่ในรูปปกติมีการกระจายตัวแบบเกาส์เซียนเราสามารถกำหนดจุดหยุดที่เป็นพื้นที่ใต้กราฟเกาส์เซียนที่เท่ากันได้ โดยสามารถหาค่าของจุดหยุดได้ในตารางสถิติตามภาพที่ 2.6



ภาพที่ 2.5 แสดงการกระจายของข้อมูลอนุกรมเวลา 8 ชุดในกราฟความน่าจะเป็นแบบปกติ ซึ่งจะเห็นว่าเป็นสมการเส้นตรงดังนั้นจึงคาดการณ์ได้ว่าข้อมูลนั้นมาจากการกระจายตัวแบบเกาส์เซียน

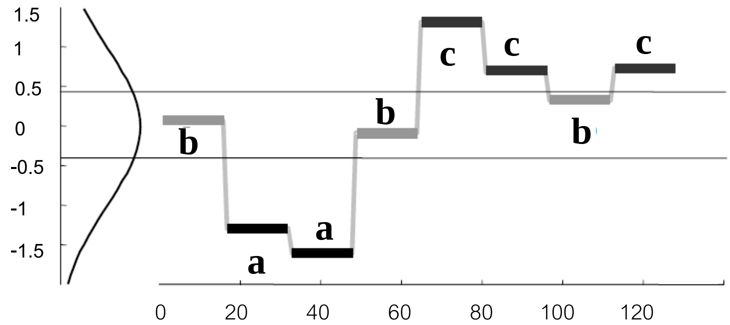
(ที่มา : Lin, J., Keogh, E., Lonardi, S. and Patel, P. Finding Motifs in Time Series. The Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining 2002)

β_i \ a	3	4	5	6	7	8	9	10
β_1	-0.43	-0.67	-0.84	-0.97	-1.07	-1.15	-1.22	-1.28
β_2	0.43	0	-0.25	-0.43	-0.57	-0.67	-0.76	-0.84
β_3		0.67	0.25	0	-0.18	-0.32	-0.43	-0.52
β_4			0.84	0.43	0.18	0	-0.14	-0.25
β_5				0.97	0.57	0.32	0.14	0
β_6					1.07	0.67	0.43	0.25
β_7						1.15	0.76	0.52
β_8							1.22	0.84
β_9								1.28

ภาพที่ 2.6 ตารางแสดงค่าของจุดหยุด โดย *a* คือจำนวนประเภทของตัวอักษรที่ใช้ในการแทนค่าให้เป็นข้อมูลวิยุต และ β_i คือ ค่าของจุดหยุดที่ใช้ในการแบ่งพื้นที่ใต้กราฟเกาส์เซียน

(ที่มา : Lin, J., Keogh, E., Lonardi, S. and Patel, P. Finding Motifs in Time Series. The Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining 2002)

หลังจากนั้นเราจึงกำหนดตัวอักษรลงไปในแต่ละส่วนของพื้นที่ใต้กราฟเกาส์เซียน ข้อมูลอนุกรมเวลา *C* จะถูกทำให้เป็นข้อมูลตัวอักษรขนาด 8 ตัวอักษรโดยมีได้ 3 ค่าคือ a, b และ c ตามภาพที่ 2.7



ภาพที่ 2.7 แสดงการแบ่งพื้นที่ใต้กราฟเกาส์เซียนซึ่งจุดแบ่งจะสามารถแยกข้อมูลอนุกรมเวลาออกเป็นส่วน ๆ ซึ่งกำหนดด้วยตัวอักษร

(ที่มา : Lin, J., Keogh, E., Lonardi, S. and Patel, P. Finding Motifs in Time Series. The Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining 2002)

จากนั้นจึงเข้าสู่ขั้นตอนสุดท้ายคือการวัดระยะระหว่างแต่ละลำดับย่อยของข้อมูลอนุกรมเวลาที่ถูกแปลงเป็นข้อมูลวิยุตที่เป็นตัวอักษรโดยใช้สมการ

$$\text{MINDIST}(Q,C) = \sqrt{\frac{n}{w}}\sqrt{\sum_{i=1}^{n}(\text{dist}(q_i, c_i))^2} \tag{2.3}$$

ฟังก์ชัน dist() จะใช้ตารางดังภาพที่ 2.8 ในการหาค่าระยะทาง ซึ่งสามารถหาได้จากสมการที่ 2.4 โดยค่าในตารางจะเป็น 0 เมื่อค่าสัมบูรณ์ของ r-c ≤ 1 โดยที่ r คือ แถวที่ และ c คือ สดมภ์ เป็นเงื่อนไขแรกและจะเท่ากับค่าสัมบูรณ์ของ $\beta_i - \beta_{j-1}$ เมื่อค่าสัมบูรณ์ของ r-c เป็นค่าที่นอกเหนือจากเงื่อนไขแรก โดย i=max(r, c)-1 และ j-1=min(r, c)

$$\text{cell}_{r,c} = \begin{cases} 0, & \text{if } |r-c| \le 1 \\ |\beta_i - \beta_{j-1}|, & \text{otherwise} \end{cases} \tag{2.4}$$

	a	b	c
a	0	0	0.86
b	0	0	0
c	0.86	0	0

ภาพที่ 2.8 ตารางแสดงค่าของระยะทางระหว่างตัวอักษรซึ่งนำไปใช้ในฟังก์ชัน dist()

(ที่มา : Lin, J., Keogh, E., Lonardi, S. and Patel, P. Finding Motifs in Time Series. The Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining 2002)

การคำนวณเพื่อเปรียบเทียบค่าระยะทางของ 2 ลำดับย่อยใด ๆ จึงทำได้โดยการเปรียบเทียบตัวอักษร 2 ชุดซึ่งเป็นตัวแทนของลำดับย่อยคู่นั้น ๆ ดังแสดงในภาพที่ 2.9

$$
\begin{array}{l}
C = \text{baabccbc} \\
\quad\ \updownarrow\updownarrow\updownarrow\updownarrow\updownarrow\updownarrow\updownarrow\updownarrow \\
Q = \text{babcacca}
\end{array}
$$

ภาพที่ 2.9 แสดงการเปรียบเทียบระหว่างตัวอักษร 2 ชุด

จากภาพที่ 2.9 จะสามารถคำนวณค่าระยะทางของแต่ละคู่ของตัวอักษร โดยใช้ตารางดังแสดงในภาพที่ 2.8 ได้ ดังนี้

$$\sum_{i=1}^{8}(\text{dist}(q_i,c_i))^2 = (0)^2+(0)^2+(0)^2+(0)^2+(0.86)^2+(0)^2+(0)^2+(0.86)^2 = 1.4792$$

ในวิธีการนี้มีจำนวนของพารามิเตอร์ที่ต้องกำหนด 2 ตัวคือ จำนวนส่วนที่เราต้องการจะแบ่ง w และจำนวนชนิดของตัวอักษร a ซึ่งยากที่จะรู้ว่าควรกำหนดแบบใดจึงจะเหมาะสมดังนั้นในงานวิจัยชิ้นนี้ จึงใช้การทดลองแล้วประมาณค่าของพารามิเตอร์ w และ a ที่ใช้ด้วยสมการ

$$\text{Tightness of Lower Bound} = \frac{\text{MINDIST}(Q,C)}{\text{D}(Q,C)} \tag{2.5}$$

โดยค่าของ D(Q, C) คือ ระยะทางยุคลิดของลำดับย่อย ซึ่งคำนวณได้ด้วยสมการ

$$\text{D}(Q,C) = \sqrt{\sum_{i=1}^{n}(q_i - c_i)^2} \tag{2.6}$$

วิธีการในส่วนของงานวิจัยชิ้นนี้ที่ได้นำเสนอมาเบื้องต้นได้รับการนำไปใช้โดยงานวิจัยรุ่นถัดมาที่พัฒนาอัลกอริทึมในการหาโมทีฟที่ความยาวแตกต่างกัน รายละเอียดส่วนอื่นของงานวิจัยชิ้นนี้ที่กล่าวถึงการทดลองเพื่อประมาณค่าของพารามิเตอร์ w และ a รวมไปถึงการพัฒนาอัลกอริทึม การทดลองและการวัดผล สามารถอ่านเพิ่มเติมได้ในงานวิจัยอ้างอิง [12]

2.2.2 **การค้นพบแบบแม่นยำของโมทีฟในข้อมูลอนุกรมเวลา** [9]

งานวิจัยชิ้นนี้เป็นงานวิจัยที่เกี่ยวกับการค้นพบโมทีฟของข้อมูลอนุกรมเวลาล่าสุดที่สามารถหาโมทีฟของข้อมูลอนุกรมเวลาที่ถูกต้องแม่นยำโดยไม่ใช่แค่การประมาณค่าใกล้เคียงและสามารถประมวลผลได้เร็วกว่าวิธีอื่นมาก ซึ่งแสดงให้เห็นในผลการทดลองกับข้อมูลทดสอบต่าง ๆ

คำจำกัดความของข้อมูลอนุกรมเวลาที่ใช้ในงานวิจัยชิ้นนี้มีดังต่อไปนี้

1. Time Series : คือ ลำดับของจำนวนจริงที่ต่อเนื่องกันเป็นจำนวน n ตัว $T = (t_1, \ldots, t_n)$

2. Time Series Database (D) : เป็นชุดของข้อมูลอนุกรมเวลาจำนวน m ชุดที่อาจมีความยาวที่แตกต่างกัน

3. Time Series Motif ของ Time Series Database (D) : คือ คู่ของข้อมูลอนุกรมเวลา $\{T_i, T_j\}$ ใน D ที่เป็นคู่ที่คล้ายกันมากที่สุดซึ่งหมายถึงมีระยะทางระหว่างคู่ข้อมูลอนุกรมเวลาน้อยที่สุด

4. k^{th}-Time Series Motif : คู่ของอนุกรมเวลาในฐานข้อมูล D ที่คล้ายกันมากที่สุดเป็นอันดับที่ k โดยคู่ของข้อมูลอนุกรมเวลา $\{T_i, T_j\}$ เป็นคู่ที่คล้ายกันมากที่สุดเป็นอันดับที่ k ก็ต่อเมื่อ มีชุดข้อมูล S ของคู่ของข้อมูลอนุกรมเวลาขนาด k-1 โดย $\forall T_d \in D$ และ $\{T_i, T_d\} \notin S$ และ $\{T_j, T_d\} \notin S$ และ $\forall \{T_x, T_y\} \in S$, $\{T_a, T_b\} \notin S$ และระยะทาง $\mathrm{dist}(T_x, T_y) \leq \mathrm{dist}(T_i, T_j) \leq \mathrm{dist}(T_a, T_b)$ ตัวอย่างเช่น $k = 5$ จะมีจำนวนคู่ของข้อมูลอนุกรมเวลาอยู่ในชุด S จำนวน 4 คู่ คือ $\{T_x, T_y\}$ ซึ่ง $\{T_i, T_j\}$ จะเป็น 5^{th}-Time Series Motif ก็ต่อเมื่อมี $\{T_x, T_y\}$ จำนวน 4 คู่ที่มีระยะทางน้อยกว่าหรือเท่ากับ $\{T_i, T_j\}$ และมี $\{T_a, T_b\}$ ที่อยู่นอกชุด S ทั้งหมดที่ระยะทางมากกว่าหรือเท่ากับ $\{T_i, T_j\}$

5. The Range Motif with Range R : คือ ชุดของข้อมูลอนุกรมเวลาที่มีคุณสมบัติว่าแต่ละข้อมูลในชุดจะมีระยะทางระหว่างกันน้อยกว่า $2R$ ซึ่งเขียนเป็นทางการได้ว่า S เป็น

Range Motif ด้วยค่า range R ก็ต่อเมื่อ $\forall T_x, T_y \in S$, dist(T_x, T_y) $\leq 2R$ และ $\forall T_d \in D\text{-}S$ dist(T_d, T_y) > $2R$ คำจำกัดความนี้ใช้เป็นตัววัดความหนาแน่นในแต่ละบริเวณของพื้นที่ของอนุกรมเวลา

คำจำกัดความที่กล่าวมาข้างต้นสามารถต่อยอดให้นำไปใช้กับลำดับย่อยของหนึ่งข้อมูลอนุกรมเวลา โดยแทนที่จะพิจารณาข้อมูลอนุกรมเวลาหลายข้อมูลก็เปลี่ยนเป็นพิจารณาลำดับย่อยทั้งหมดของหนึ่งข้อมูลอนุกรมเวลาในฐานข้อมูลอนุกรมเวลา D แทน โดยมีคำจำกัดความต่อเนื่องดังนี้

6. Subsequence : ลำดับย่อยขนาดความยาว n ของข้อมูลอนุกรมเวลา $T = (t_1, t_2 \ldots, t_m)$ คือ ข้อมูลอนุกรมเวลา $T_{i,n} = (t_i, t_{i+1}, \ldots, t_{i+n-1})$ โดยที่ $1 \leq i \leq m\text{-}n+1$

7. Subsequence Motif : เป็นคู่ของลำดับย่อย $\{T_{i,n}, T_{j,n}\}$ ของข้อมูลอนุกรมเวลาขนาดความยาว T ที่มีความคล้ายกันมากที่สุด คือระยะทางของคู่ลำดับย่อยทั้งหมดมีค่าน้อยกว่าระยะทางของคู่ $\{T_{i,n}, T_{j,n}\}$

อัลกอริทึมที่งานวิจัยนี้ได้นำเสนอมีชื่อว่า Mueen-Keogh (MK) ซึ่งเป็นวิธีการค้นพบโมทีฟของข้อมูลอนุกรมเวลาโดยใช้หลักการทางคณิตศาสตร์เรื่องคุณสมบัติของความไม่เท่ากันของสามเหลี่ยมในการข้ามการคำนวณระยะทางที่ไม่สำคัญทิ้งไป โดยวิธีการของอัลกอริทึมจะเริ่มต้นด้วยการกำหนดลำดับย่อยแบบสุ่มขึ้นมาชุดหนึ่งซึ่งเรียกว่าเป็นตัวอ้างอิง กำหนดให้เป็นชุด $S = \{s_1, \ldots, s_n\}$ โดย n เป็นจำนวนตัวอ้างอิงที่สุ่มขึ้นมาจากข้อมูลอนุกรมเวลา $T = \{t_1, \ldots, t_m\}$ โดย m คือจำนวนลำดับย่อยทั้งหมดที่เป็นไปได้ใน T แล้วทำการคำนวณหาระยะทางยุคลิดของตัวอ้างอิงกับลำดับย่อยที่เป็นไปได้ทั้งหมดใน T นั้น เมื่อสังเกตที่ตัวอ้างอิง 1 ตัวที่ s_1 เราจะได้ระยะทางจาก s_1 ไปยังลำดับย่อย t_1 *ถึง* t_m โดยที่ระยะทางที่น้อยที่สุดจะถูกเก็บไว้เป็นตัวแปรที่ชื่อว่า *Best-so-far* (*BSF*) สมมติว่าเราพิจารณาที่ระยะทางของคู่ลำดับย่อยโดยให้ A = dist(s_1, t_1), B = dist(s_1, t_2) และ C = dist(t_1, t_2) ค่าของ A และ B นี้เป็นเหมือนค่าของด้าน 2 ด้านของสามเหลี่ยม ซึ่งเราสามารถประมาณค่าของ C ได้จากอสมการความไม่เท่ากันของสามเหลี่ยม คือ $|A\text{-}B| \leq C$ ดังนั้น เราจะได้ค่า $|A\text{-}B|$ เป็นขอบเขตล่างของ C หมายถึงเป็นค่าประมาณที่ไม่เกินค่าของ C ซึ่งถ้าค่าขอบเขตล่างนี้มากกว่า *BSF* แล้วแสดงว่า dist(t_1, t_2) ต้องมากกว่า *BSF* ด้วยแน่นอน เราจึงสามารถทำการข้ามการคำนวณระยะทางยุคลิดของ t_1 และ t_2 ไปได้เลยเพราะคู่นี้ไม่มีโอกาสที่จะเป็นคู่โมทีฟอีกแล้วเพราะระยะทางมากกว่าซึ่งหมายถึงความคล้ายกันจึงน้อยกว่าด้วย

อัลกอริทึมของงานวิจัยนี้จึงเป็นอัลกอริทึมหนึ่งที่เหมาะสมในการนำมาใช้หาคู่โมทีฟของข้อมูลอนุกรมเวลาเพื่อการศึกษาในการหาความยาวของโมทีฟ

2.2.3 การค้นพบโมทีฟต้นแบบที่ความยาวแตกต่างกันในข้อมูลอนุกรมเวลา [10]

งานวิจัยที่เกี่ยวข้องกับการค้นพบโมทีฟความยาวแปรผันในข้อมูลอนุกรมเวลา โดยในงานชิ้นนี้ได้กล่าวถึงปัญหาของการกำหนดความยาวโมทีฟและได้นำเสนออัลกอริทึมใหม่ในการค้นพบโมทีฟโดยไม่จำเป็นต้องกำหนดความยาวที่เรียกว่า Motif Concatenation

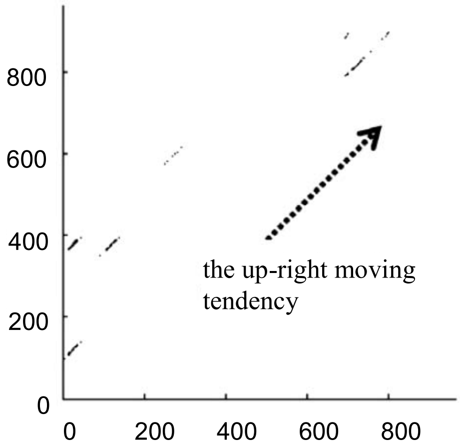
คำจำกัดความที่ใช้ในงานวิจัยมีดังต่อไปนี้

1. Time Series : ข้อมูลอนุกรมเวลา $T = (t_1, t_2, \ldots, t_m)$ เป็นลำดับจำกัดของจำนวนจริง โดย m คือความยาวของข้อมูลอนุกรมเวลา

2. Collision Matrix : เมทริกซ์การชนกัน *CM* ของข้อมูลอนุกรมเวลา T คือ เมทริกซ์ขนาด $q \times q$ โดยที่ q คือ จำนวนลำดับย่อยของข้อมูลอนุกรมเวลา T โดยสมาชิกแต่ละตัวบนเมทริกซ์ (M) แสดงด้วย $e_{ij} \in M$, e_{ij} = collision_hit (s_i, s_j) โดย collision_hit คือ ดีกรีความคล้ายกันของ 2 ลำดับย่อยที่ได้มาจากการสร้างเมทริกซ์การชนกันด้วยวิธีการฉายแบบสุ่ม (Random Projection) [3] โดยช่องใดที่มีดีกรีการชนกันสูงสุดจะเรียกว่าเป็น top-*k* Matches ในที่นี้ลำดับย่อยคู่ใดที่ช่องบนเมทริกซ์มีดีกรีการชนกันสูงสุดจะเป็น โมทีฟของข้อมูลอนุกรมเวลา

3. Motif : สำหรับลำดับย่อยใด ๆ 2 ลำดับ s_i, $s_j \in S$ ด้วยความยาว w ถ้าดีกรีการชนกัน e_{ij} = collision_hit(s_i, s_j) มีดีกรีการชนกันสูงสุดเป็นอันดับที่ k เราเรียก คู่ของ $M = (s_i, s_j)$ เป็น *k*-motif ของข้อมูลอนุกรมเวลา T โดย k คือ อันดับของดีกรีการชนกัน

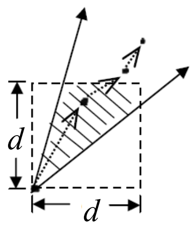
อัลกอริทึมของงานวิจัยชิ้นนี้ใช้การสร้างเมทริกซ์การชนกันด้วยวิธี Random Projection [3] เพื่อหาโมทีฟที่ความยาวแตกต่างกันแล้วนำแต่ละข้อมูลที่เป็นจุดแสดงตำแหน่งของโมทีฟของข้อมูลอนุกรมเวลามาต่อกันซึ่งแสดงได้ดังภาพที่ 2.10 ซึ่งวิธีนี้จะต้องกำหนดพารามิเตอร์จำนวน 3 ตัว คือ จำนวนเซกเมนต์ s และจำนวนของประเภทตัวอักษร n เพื่อแปลงแต่ละลำดับย่อยของข้อมูลอนุกรมเวลา T ให้เป็นข้อมูลวิยุต และความยาวเริ่มต้น w จะต้องถูกกำหนดให้ใกล้เคียงกับรูปแบบที่น่าสนใจในข้อมูลอนุกรมเวลา



ภาพที่ 2.10 เมทริกซ์การชนกันที่แสดงตำแหน่งของโมทีฟของข้อมูลอนุกรมเวลา

(ที่มา : Tang, H. and Liao, S. S. Discovering original motifs with different lengths from time series. Knowledge-Based Systems 2008: 666–671)

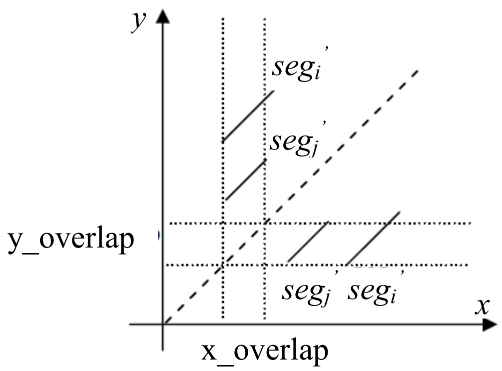
จากนั้นจะทำการแบ่งจุดแสดงตำแหน่งของโมทีฟบนเมทริกซ์การชนกันแต่ละจุดออกเป็นกลุ่ม ๆ เรียกว่า เซกเมนต์ โดยวิธีการแบ่งจุดนั้นทำได้โดยการกำหนดพารามิเตอร์เพิ่มเติมอีก 3 ตัวคือ d, α_1 และ α_2 โดยที่ d คือ ระยะทางที่จะใช้ต่อจุดของโมทีฟ ส่วน α_1 และ α_2 คือ ความชันของเส้นตรง 2 เส้นที่เป็นกรอบในการต่อจุดซึ่งแสดงไว้ดังภาพที่ 2.11



ภาพที่ 2.11 อัลกอริทึม Motif Concatenation ซึ่งต้องกำหนดพารามิเตอร์ 3 ตัวคือ d คือ ระยะการต่อจุดโมทีฟ α_1 และ α_2 คือ ความชันของเส้นตรง 2 เส้นที่กำหนดกรอบการต่อจุด

(ที่มา : Tang, H. and Liao, S. S. Discovering original motifs with different lengths from time series. Knowledge-Based Systems 2008: 666–671)

หลังจากการต่อจุดแล้วจะได้ผลดังภาพที่ 2.10 จากนั้นความยาวโมทีฟบนจุดของส่วนที่เกยทับกันของแต่ละเซกเมนต์จะนำมาคำนวณค่าเฉลี่ยเพื่อเป็นขนาดของความยาวโมทีฟที่เป็นคำตอบของอัลกอริทึม ซึ่งแสดงการเกยทับกันไว้ดังภาพที่ 2.12



ภาพที่ 2.12 ส่วนที่เกยทับกันตามรูปจากแกน *x* ซึ่งมีการฉายลงบนแกน *y* เพื่อตรวจสอบดีกรีการเกยทับกันของทั้งสองแกนตามคำจำกัดความในงานวิจัยซึ่งอ่านเพิ่มเติมได้ใน [10]
(ที่มา : Tang, H. and Liao, S. S. Discovering original motifs with different lengths from time series. Knowledge-Based Systems 2008: 666–671)

ในงานวิจัยนี้ยังต้องกำหนดค่าความยาวเริ่มต้น *w* เพื่อแปลงให้แต่ละลำดับย่อยเปลี่ยนเป็นข้อมูลวิยุตก่อนเข้าสู่ขั้นตอน Random Projection ซึ่งผู้ใช้งานต้องเข้าใจวิธี Random Projection เพื่อกำหนดพารามิเตอร์ 2 ตัว โดยรวมแล้วอัลกอริทึมนี้ยังมีความซับซ้อนของการใช้งานเนื่องจากต้องกำหนดพารามิเตอร์ 6 ตัวคือ *s*, *n*, *d*, α_1 ,α_2 และ ค่าความยาวเริ่มต้น *w* ที่กล่าวไว้เบื้องต้น โดย *d*, α_1 และ α_2 ต้องพิจารณาจากเมทริกซ์การชนกัน (*CM*) ที่สร้างขึ้นว่าควรให้ค่าเป็นเท่าไร อีกทั้งวิธีการต่อจุดของอัลกอริทึม Motif Concatenation ยังต้องการใช้เมทริกซ์การชนกันเพื่อสร้างพื้นที่และความชันตามพารามิเตอร์ที่ใส่เข้าไป ดังนั้น เมทริกซ์การชนกัน (*CM*) จึงต้องเป็นข้อมูลนำเข้าอีกหนึ่งตัว ซึ่งมีความซับซ้อนในการใช้งาน อีกทั้งยังต้องกำหนดพารามิเตอร์ความยาวเริ่มต้น *w* ในอัลกอริทึมซึ่งไม่แตกต่างกับการกำหนดความโมทีฟเนื่องจากต้องพิจารณาข้อมูลอนุกรมเวลาโดยรวมเหมือนเดิมว่าควรกำหนดความยาวเริ่มต้นเป็นเท่าไร ส่วนของผลลัพธ์ที่ได้จากอัลกอริทึม Motif Concatenation เป็นลำดับย่อยที่ต้องนำมาพิจารณาอีกว่าส่วนใดเป็นโมทีฟจึงต้องอาศัยการวิเคราะห์จากผู้ใช้งานอีกครั้งหนึ่งแล้วจึงตัดส่วนที่ไม่ต้องการออกโดยผู้ใช้งาน โมทีฟที่ได้จึงแตกต่างกันออกไปตามผู้ใช้งานด้วยว่าจะเลือกตัดส่วนใด

2.2.4 การค้นพบโมทีฟในข้อมูลอนุกรมเวลาที่ความยาวแปรผันโดยประมาณด้วยวิธีการเหนี่ยวนำไวยากรณ์ [11]

งานวิจัยชิ้นนี้เป็นงานวิจัยที่มีวัตถุประสงค์ในการหาโมทีฟโดยไม่ต้องกำหนดความยาวของข้อมูลอนุกรมเวลาอีกงานหนึ่งที่ใช้หลักการของ Grammar Induction โดยมีคำจำกัดความที่ใช้ในงานวิจัย ดังนี้

1. Time Series : ข้อมูลอนุกรมเวลา $T = t_1, t_2, \ldots, t_m$ คือชุดของลำดับของจำนวนจริงจำนวน m ตัว

2. Subsequence : ถ้ามีข้อมูลอนุกรมเวลา T ขนาดความยาว m แล้วลำดับย่อย C ใน T คือส่วนแบ่งย่อยความยาว $n \leq m$ ที่ตำแหน่งที่ติดกันจาก p เป็นต้นไป นั่นคือ $C = t_p, \ldots, t_{p+n-1}$ โดย $1 \leq p \leq m-n+1$

3. Sliding Window : ถ้ามีข้อมูลอนุกรมเวลา T และความยาวของลำดับย่อยที่ผู้ใช้งานกำหนดขนาด n แล้วทุก ๆ ลำดับย่อยที่เป็นไปได้ของ T จะถูกแยกออกมาได้โดย sliding window ขนาดความยาว n เลื่อนผ่าน T โดยพิจารณาเฉพาะลำดับย่อย C_p ที่ $1 \leq p \leq m-n+1$

อัลกอริทึมที่นำมาใช้มีชื่อว่า Sequitur ซึ่งเป็นวิธีที่ต้องใช้บนข้อมูลวิยุตดังนั้นจึงเริ่มต้นด้วยการทำให้ข้อมูลอนุกรมเวลาเปลี่ยนเป็นข้อมูลวิยุตโดยใช้ SAX – Symbolic Aggregate approXimation [13] ซึ่งมีพารามิเตอร์ที่ต้องกำหนด 2 ตัว คือ จำนวนเซกเมนต์ s และจำนวนของประเภทตัวอักษร n แล้วจึงประมวลผลข้อมูลตัวอักษรนั้นด้วย Sequitur ซึ่งต้องกำหนดพารามิเตอร์ 1 ตัวคือ i เป็นค่าความยาวโมทีฟเริ่มต้นค่าน้อย ๆ ซึ่งต้องประมาณค่าโดยผู้ใช้งาน ผลลัพธ์ของอัลกอริทึมจะหารูปแบบของตัวอักษรที่ซ้ำกัน เรียกว่า กฎของไวยากรณ์ (Grammar Rule) ซึ่งเป็นโมทีฟจากนั้นจึงเทียบตำแหน่งของข้อมูลวิยุตที่เป็นตัวอักษรในกฎของไวยากรณ์กลับไปหาตำแหน่งของโมทีฟบนข้อมูลอนุกรมเวลา

อัลกอริทึมนี้มีพารามิเตอร์ที่จะต้องทำการกำหนดเพิ่ม 3 ตัว ตามที่ได้กล่าวไว้ข้างต้น คือ ความยาวเริ่มต้นของโมทีฟซึ่งเป็นค่าที่ป้องกันไม่ให้โมทีฟที่เกิดขึ้นมีขนาดเล็กเกินไป โดยจะปรับความยาวเพิ่มขึ้นไปเรื่อย ๆ เพื่อหาความยาวที่เป็นคำตอบ การนำวิธีการของ SAX มาใช้ในอัลกอริทึมจึงต้องกำหนดพารามิเตอร์เพิ่มขึ้นอีก 2 ตัว คือ จำนวนเซกเมนต์ ซึ่งใช้ในการแบ่งส่วนข้อมูลอนุกรมเวลาเพื่อลดจำนวนมิติลงและจำนวนของตัวอักษรที่จะใช้แทนข้อมูลอนุกรมเวลา อัลกอริทึมนี้ยังมีความซับซ้อนในการใช้งานเพราะผู้ใช้ต้องเข้าใจอัลกอริทึม SAX เพื่อใช้กำหนดพารามิเตอร์และต้องการการวิเคราะห์จากผู้ใช้ว่าควรจะให้ค่าของความยาวเริ่มต้นค่าน้อย ๆ เป็นเท่าไร ซึ่งแทบไม่แตกต่างกับการกำหนดความยาวโมทีฟปกติ

เนื่องจากงานวิจัยที่ผ่านมาที่ได้นำเสนอวิธีการหาโมทีฟโดยไม่ต้องกำหนดค่าความยาวยังมีความซับซ้อนในการใช้งานโดยต้องกำหนดพารามิเตอร์เพิ่มขึ้นจำนวนหนึ่ง โดยเฉพาะอย่างยิ่งค่าของความยาวเริ่มต้น w และในแต่ละขั้นตอนยังต้องการการวิเคราะห์จาก

ผู้ใช้งานด้วย ผลลัพธ์ที่ได้ยังไม่มีความแน่ชัดในเรื่องของคุณภาพของโมทีฟที่ได้ว่าเป็นอย่างไร ดังนั้นในงานวิจัยของวิทยานิพนธ์ฉบับนี้จึงพัฒนาอัลกอริทึมในการหาโมทีฟเพื่อแก้ปัญหาของงานวิจัยที่ผ่านมา รวมไปถึงการประเมินคุณภาพโมทีฟที่ชัดเจนมากขึ้น

บทที่ 3

การค้นพบโมทีฟความยาวแปรผันสำหรับข้อมูลอนุกรมเวลา

เมื่อกล่าวถึงการพัฒนาวิธีการค้นพบโมทีฟในความยาวที่เหมาะสมนั้น คงหลีกเลี่ยงไม่ได้ที่จะเริ่มต้นการพัฒนาด้วยการหาโมทีฟที่ความยาวแตกต่างกันทั้งหมดที่เป็นไปได้ซึ่งมีหลายค่า เพื่อศึกษาลักษณะของโมทีฟที่เกิดขึ้น ดังนั้น ขั้นตอนแรกจำเป็นต้องมีอัลกอริทึมในการค้นพบโมทีฟที่มีประสิทธิภาพ โดยในงานวิจัยนี้เลือกใช้อัลกอริทึม MK (Mueen-Keogh) [9] ซึ่งเป็นอัลกอริทึมที่ใช้ในการหาโมทีฟที่ดีที่สุดในขณะนี้ จากนั้น ในขั้นตอนถัดไปจะเป็นการวิเคราะห์โมทีฟที่ค้นพบในแต่ละความยาวของข้อมูลอนุกรมเวลา แล้วจึงทำการลดจำนวนโมทีฟที่ได้ทั้งหมดลงจนกระทั่งได้เซตของ “โมทีฟที่ดี” เป็นผลลัพธ์ โดยมีขั้นตอน ดังต่อไปนี้ ขั้นตอนแรกจะแบ่งกลุ่มโมทีฟโดยใช้เกณฑ์ความคล้ายกันของตำแหน่งที่โมทีฟถูกค้นพบและเกณฑ์ขอบเขตบนของจำนวนโมทีฟ ขั้นตอนที่สองจะทำการเลือกตัวแทนของโมทีฟจากกลุ่มที่เกิดจากการแบ่งกลุ่มนั้น จากนั้นตัวแทนของโมทีฟแต่ละกลุ่มจะนำมาทำการรวมกลุ่มใหม่แล้วคำนวณคะแนนเพื่อทำการจัดอันดับหากลุ่มที่ดีที่สุด และในขั้นตอนสุดท้ายจะทำการหาตัวแทนของแต่ละกลุ่มนั้นออกมาเป็นคำตอบของอัลกอริทึม

ลำดับการนำเสนอในบทที่ 3 นี้ จะเริ่มต้นจากขั้นตอนการหาโมทีฟที่ความยาวแตกต่างกันซึ่งเป็นขั้นตอนแรกของวิธีการ จากนั้นจะดำเนินเรื่องตามขั้นตอนของการพัฒนาที่กล่าวไว้เบื้องต้น คือ การแบ่งกลุ่มโมทีฟ การหาตัวแทนของกลุ่มโมทีฟ การคำนวณคะแนนของกลุ่มตัวแทนโมทีฟเพื่อจัดอันดับ “โมทีฟที่ดี” ออกมาเป็นเซตคำตอบของอัลกอริทึม

3.1 คำจำกัดความที่ใช้ในงานวิจัย

1. ข้อมูลอนุกรมเวลา (Time Series) คือ ลำดับของจำนวนจริงที่ต่อเนื่องกันที่จำนวน n ตัว $T = (t_1, t_2, \ldots, t_n)$

2. ลำดับย่อย (Subsequence) S_i^w คือ ลำดับของจำนวนจริงต่อเนื่องกันที่มีขนาดสั้นกว่าในข้อมูลอนุกรมเวลา T โดย w คือ ขนาดความยาวของลำดับย่อยและ i คือตำแหน่งเริ่มต้นของลำดับย่อยนั้น โดยลำดับย่อย $S_i^w = (t_i, t_{i+1}, \ldots, t_{i+w-1})$ โดย $w > 1$

3. ระยะทางยุคลิด EUC(S_i^w, S_j^w) คือ ระยะทางระหว่าง 2 ลำดับย่อย S_i^w และ S_j^w ของข้อมูลอนุกรมเวลา T โดย $\text{EUC}(S_i^w, S_j^w) = \sqrt{\sum_{k=1}^{w}(t_{k+i-1} - t_{k+j-1})^2}$ โดย $1 \leq i \leq n\text{-}w$ และ $1 \leq j \leq n\text{-}w$

4. โมทีฟของข้อมูลอนุกรมเวลา (Time Series Motif) M_w คือ คู่ของลำดับย่อยที่มีรูปร่างคล้ายกันในข้อมูลอนุกรมเวลา T ที่ขนาดความยาว w กำหนดโดย $M_w = (S_{L_1}{}^w, S_{L_2}{}^w$, $MDist$, $NDist)$ เมื่อ L_1 และ L_2 คือ ตำแหน่งเริ่มต้นของลำดับย่อย โดยที่ $L_1 < L_2$ และ $MDist = \text{EUC}(S_{L_1}{}^w, S_{L_2}{}^w)$ และ $NDist$ คือ ระยะทางยุคลิดในรูปปกติระหว่างลำดับย่อย $(S_{L_1}{}^w, S_{L_2}{}^w)$ ซึ่งคำนวณได้โดย $NDist = \frac{MDist}{w}$

5. โมทีฟของข้อมูลอนุกรมเวลาอันดับที่ k (k^{th}-Time Series Motif) k^{th}-M_w คือ โมทีฟที่มีคู่ของลำดับย่อยของอนุกรมเวลา (S_i^w, S_j^w) ที่ขนาดความยาว w ที่คล้ายกันมากที่สุดเป็นอันดับที่ k ในข้อมูลอนุกรมเวลา T โดย k^{th}-Motif และ i^{th}-Motif ไม่ซ้อนทับกันและ $1 \leq i < k$

เมื่อการซ้อนทับกันของโมทีฟอาจทำให้เกิด trivial matches [3][12] ดังนั้น การสกัดโมทีฟทั้งหมดจึงทำโดยการหาโมทีฟที่ไม่ซ้อนทับกัน

6. โมทีฟซ้อนทับกัน (Overlapped Motifs) โมทีฟสองโมทีฟจะซ้อนทับกันก็ต่อเมื่อ ($M_i.L_1 \leq M_j.L_1 \leq M_i.L_1 + i$ หรือ $M_j.L_1 \leq M_i.L_1 \leq M_j.L_1 + j$) และ ($M_i.L_2 \leq M_j.L_2 \leq M_i.L_2 + i$ หรือ $M_j.L_2 \leq M_i.L_2 \leq M_j.L_2 + j$) โดยที่ i และ j คือความยาวของโมทีฟ

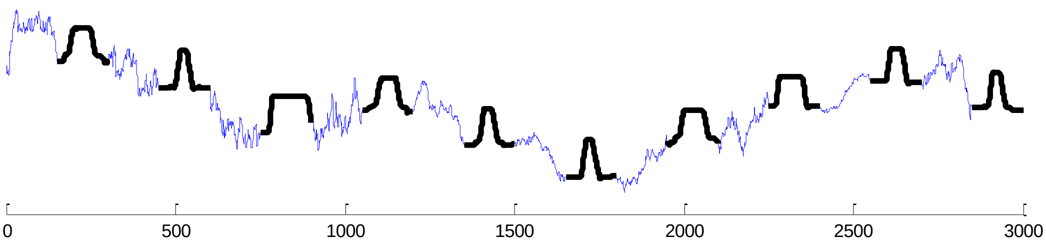
7. โมทีฟที่ดี (Best Motifs) BM_w คือ คู่ของลำดับย่อย (S_i^w, S_j^w) ที่มีรูปร่างคล้ายกันที่ยาวที่สุด ที่ถูกค้นพบที่ตำแหน่งใดตำแหน่งหนึ่ง โดยที่ EUC(S_i^w, S_j^w) $\leq R$ และ R คือ ระยะทางยุคลิดที่มากที่สุดที่ใช้ในการพิจารณาความเป็นโมทีฟที่ดี

8. โมทีฟที่ดีอันดับที่ k (k^{th}-Best Motif) k^{th}- BM_w คือ โมทีฟที่มีคะแนนสูงสุดเป็นอันดับที่ k ที่ได้จากฟังก์ชันการให้คะแนนที่นำเสนอในอัลกอริทึมซึ่งคำนวณโดยมีพื้นฐานมาจากความคล้ายกันของตำแหน่งการเกิดโมทีฟและความคล้ายกันของคู่ลำดับย่อยของโมทีฟ

3.2 การค้นพบโมทีฟของข้อมูลอนุกรมเวลา

การพัฒนาวิธีการค้นหาโมทีฟในความยาวที่เหมาะสมนั้น จำเป็นต้องทำการวิเคราะห์โมทีฟที่เกิดขึ้นที่ความยาวทั้งหมดที่เป็นไปได้ก่อน เพื่อที่จะนำลักษณะและตำแหน่งการ

เกิดของโมทีฟมาเปรียบเทียบเพื่อใช้ในการแบ่งกลุ่ม ในงานวิจัยชิ้นนี้ได้เลือกใช้อัลกอริทึม MK (Mueen-Keogh) [9] ซึ่งเป็นอัลกอริทึมที่ใช้ในการค้นพบโมทีฟที่ดีที่สุดในขณะนี้ มาใช้ค้นหาโมทีฟที่ความยาวแตกต่างกัน เพื่อความเข้าใจยิ่งขึ้น ผู้วิจัยจึงทำการสาธิตกับข้อมูลทดลองตัวอย่างซึ่งแสดงดังภาพที่ 3.1

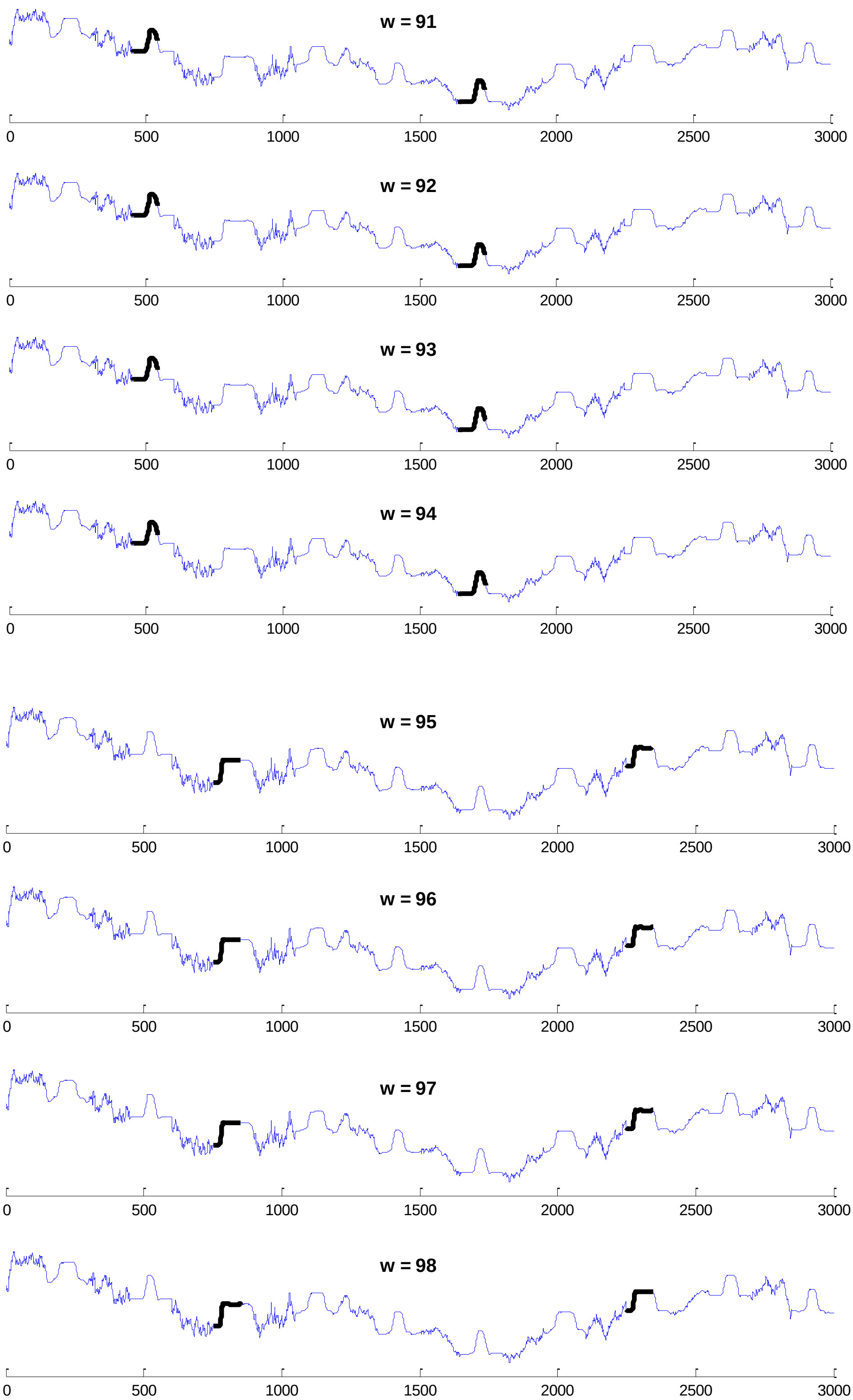


ภาพที่ 3.1 ตัวอย่างข้อมูลอนุกรมเวลา Gun-Point [1] ซึ่งสร้างจากการฝังตัวข้อมูล Gun-Point ขนาด 150 จุดข้อมูลลงไปในข้อมูลแบบสุ่ม (Random Walk) ซึ่งใช้เป็นตำแหน่งของโมทีฟที่อัลกอริทึมควรจะให้ผลลัพธ์ของโมทีฟมาที่ตำแหน่งและความยาวนี้ โดยโมทีฟที่นำมาฝังตัวจะแสดงด้วยเส้นหนา

เนื่องจากข้อมูลอนุกรมเวลา Gun-Point มีขนาด 3000 จุดข้อมูล ดังนั้น ความยาวโมทีฟ (*w*) ที่เป็นไปได้ทั้งหมด คือ *w* = 2, 3, 4, …, 1500 จุดข้อมูล โดยที่ความยาวสูงสุดคือ 1500 จุดข้อมูล จะได้คู่ของโมทีฟที่แบ่งข้อมูลอนุกรมเวลาออกเป็น 2 ส่วนเท่ากันพอดี ซึ่งผู้วิจัยได้ทำการหาโมทีฟโดยเปลี่ยนพารามิเตอร์ความยาวของอัลกอริทึม MK โดยให้ค่าตั้งแต่ 2 เป็นต้นไปจนถึง 1500 แล้ววิเคราะห์โมทีฟที่ได้จากอัลกอริทึม ผลจากการหาโมทีฟที่เกิดขึ้นในช่วงระยะความยาวหนึ่งจะพบว่าโมทีฟที่ถูกค้นพบจะเกิดในตำแหน่งที่ใกล้เคียงกัน และเมื่อความยาวเพิ่มขึ้นเรื่อย ๆ จนกระทั่งมีคู่ของลำดับย่อยที่คล้ายกันที่ตำแหน่งอื่นที่คล้ายกันมากกว่า โมทีฟที่เกิดขึ้นจะมีการเปลี่ยนแปลงตำแหน่ง โดยมีผลลัพธ์ที่ได้จากการหาโมทีฟในแต่ละความยาว ดังแสดงในภาพที่ 3.2 ในช่วงความยาวของโมทีฟตั้งแต่ 91 จุดข้อมูล ไปจนถึง 98 จุดข้อมูล

จากผลลัพธ์โมทีฟ ดังภาพที่ 3.2 สังเกตได้ว่าโมทีฟที่เกิดขึ้นในแต่ละขนาดความยาวมีทั้งโมทีฟเกิดที่ตำแหน่งใกล้เคียงกันและไม่ใกล้เคียงกัน โดยลักษณะโมทีฟที่เกิดขึ้นที่ตำแหน่งใกล้เคียงกันนั้นจะเกิดขึ้นที่ความยาวไล่เรียงกันไปช่วงหนึ่ง คือ โมทีฟที่ความยาว 91 ถึง 94 หลังจากนั้น โมทีฟจะเกิดการเปลี่ยนตำแหน่งไปจากเดิม เป็นโมทีฟที่ความยาว 95 ถึง 98 ดังนั้น จากการสังเกตลักษณะที่เกิดขึ้นเทียบกับข้อมูล Gun-Point ที่ทำการฝังตัวลงไป โมทีฟในตำแหน่งที่ใกล้เคียงกันนี้จะเป็นส่วนหนึ่งของข้อมูล Gun-Point ซึ่งเป็นข้อสังเกตได้ว่าโมทีฟที่เกิดในตำแหน่งที่

ใกล้เคียงกันจะเป็นส่วนหนึ่งของรูปแบบที่น่าสนใจ ณ ตำแหน่งนั้น โดยจะต้องมีโมทีฟที่มีขนาด



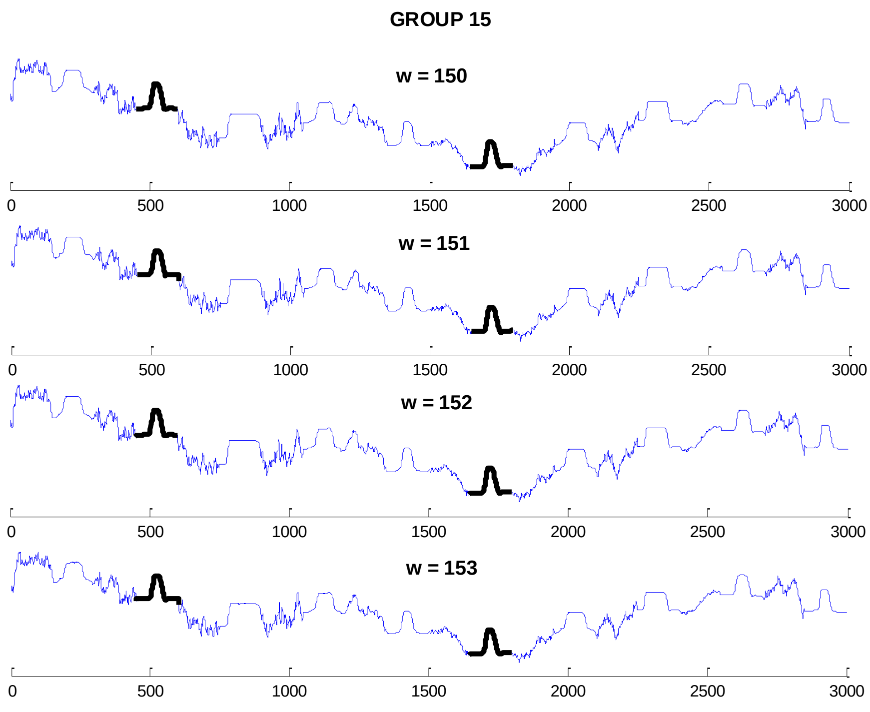
ภาพที่ 3.2 ตัวอย่างผลลัพธ์โมทีฟที่ถูกค้นพบที่ความยาวแตกต่างกัน

ความยาวค่าหนึ่งที่ครอบคลุมส่วนที่น่าสนใจทั้งหมด ณ ตำแหน่งนั้น และเนื่องจากในงานวิจัยนี้ใช้ระยะทางยุคลิดเป็นมาตรวัดความคล้ายกันของคู่ลำดับย่อย ความยาวที่เพิ่มขึ้นของคู่ลำดับย่อยจึงหมายถึงระยะทางยุคลิดที่เพิ่มขึ้น ดังนั้น ผู้วิจัยจึงต้องหาค่าที่มากสุดของระยะทางยุคลิดของคู่ลำดับย่อยที่เหมาะสมที่บอกว่า "โมทีฟที่ดี" ต้องมีค่าความคล้ายกันไม่เกินค่านี้ ซึ่งหมายถึงค่า R ที่เป็นค่าระยะทางยุคลิดที่มากที่สุดสำหรับการพิจารณาว่าเป็น "โมทีฟที่ดี" ผู้วิจัยจึงได้เริ่มต้นโดยทำการแบ่งกลุ่มของโมทีฟในแต่ละความยาวออกเป็นกลุ่ม ๆ โดยใช้เกณฑ์ความคล้ายกันของตำแหน่งที่เกิดโมทีฟ ซึ่งจะใช้การซ้อนทับกันของโมทีฟเป็นเกณฑ์ความคล้ายกันในการแบ่งกลุ่ม

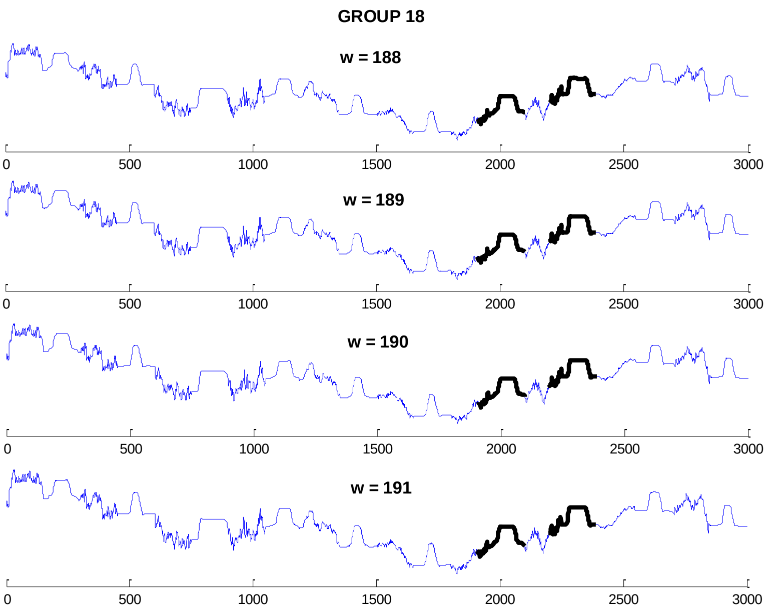
เมื่อได้ผลลัพธ์โมทีฟที่ความยาวทั้งหมดที่เป็นไปได้แล้ว ผู้วิจัยได้ทำการแบ่งกลุ่มของโมทีฟโดยใช้การซ้อนทับกันของแต่ละโมทีฟเป็นเกณฑ์ โดยโมทีฟที่แต่ละขนาดความยาวจะถูกจัดให้อยู่ในกลุ่มเดียวกันก็ต่อเมื่อคู่ลำดับย่อยของโมทีฟนั้นเกิดการซ้อนทับกัน ดังคำจำกัดความ

โมทีฟซ้อนทับกัน (Overlapped Motifs) โมทีฟสองโมทีฟจะซ้อนทับกันก็ต่อเมื่อ ($M_i.L_1 \leq M_j.L_1 \leq M_i,L_1+ i$ หรือ $M_j.L_1 \leq M_i.L_1 \leq M_j,L_1+ j$) และ ($M_i.L_2 \leq M_j.L_2 \leq M_i,L_2+ i$ หรือ $M_j.L_2 \leq M_i.L_2 \leq M_j,L_2+ j$) โดยที่ i และ j คือความยาวของโมทีฟ

ซึ่งจะได้ตัวอย่างของกลุ่มโมทีฟ ดังแสดงในภาพที่ 3.3 และภาพที่ 3.4



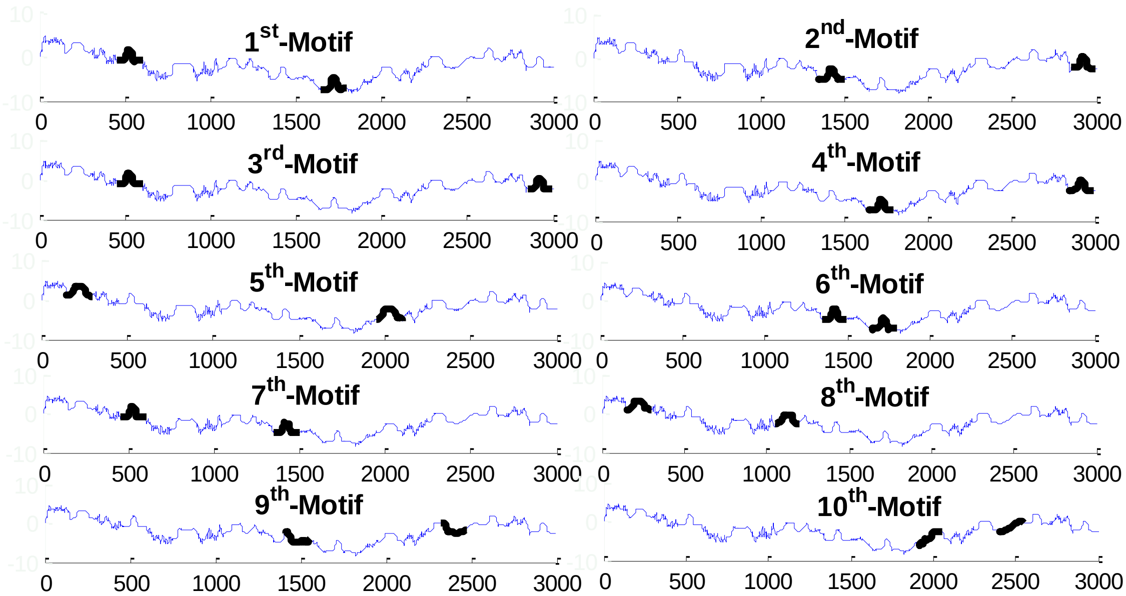
ภาพที่ 3.3 ตัวอย่างกลุ่มโมทีฟกลุ่มที่ 15 ที่เกิดขึ้นจากการแบ่งกลุ่มแบบใช้เกณฑ์การซ้อนทับกัน โดยโมทีฟที่ความยาว (w) 150, 151, 152, และ 153 ซ้อนทับกัน



ภาพที่ 3.4 ตัวอย่างกลุ่มโมทีฟกลุ่มที่ 18 ที่เกิดขึ้นจากการแบ่งกลุ่มแบบใช้เกณฑ์การซ้อนทับกัน

สังเกตว่ารูปแบบของโมทีฟที่ได้ในกลุ่มที่ 15 จะมีกลุ่มที่มีโมทีฟความยาว 150 จุดข้อมูล ซึ่งเป็นโมทีฟที่ขนาดความยาวเดียวกับโมทีฟที่ทำการฝังตัว ซึ่งได้โมทีฟออกมารูป แบบตรงกับโมทีฟที่ได้ฝังตัวลงไปในข้อมูลทดลอง ถัดมาในโมทีฟกลุ่มที่ 18 จะพบว่ารูปแบบและตำแหน่งของโมทีฟนั้นใกล้เคียงกับโมทีฟที่ทำการฝังตัวลงไป แต่จะมีข้อมูลแบบสุ่มรวมออกมาด้วย จากผลการทดลองเบื้องต้นของตัวอย่างนี้แสดงให้เห็นว่า การหาโมทีฟที่ความยาวทั้งหมดที่เป็นไปได้นั้นยังไม่เพียงพอที่จะสามารถค้นพบโมทีฟที่ฝังตัวลงไปได้ทั้งหมด เนื่องจากโมทีฟที่เป็นโมทีฟที่ดีในตัวอย่างข้อมูลทดลองนี้มีขนาดความยาวเท่ากัน คือ 150 จุดข้อมูล วิธีการค้นพบเพียงหนึ่งโมทีฟต่อหนึ่งความยาวจึงไม่เพียงพอเพราะอาจมีคู่โมทีฟอื่นที่ความยาวเดียวกันเหลืออยู่แต่ความคล้ายของโมทีฟนั้นเทียบระยะทางยุคลิดแล้วมากกว่าโมทีฟคู่แรกที่ค้นพบ ดังนั้น ผู้วิจัยจึงได้ทำการเพิ่มการค้นหาโมทีฟของแต่ละความยาว โดยนำ k^{th}-Time Series Motif มาใช้ในงานวิจัย โดยจะเริ่มต้นหา k^{th}-Motif ที่เป็นไปได้ในแต่ละความยาว โดยวิธีการหา k^{th}-Motif ที่เป็นไปได้นั้น ผู้วิจัยได้ทำการพัฒนาโดยแยกออกเป็น 2 วิธีการ ดังต่อไปนี้

1. **ค้นหาโมทีฟทั้งหมดที่เป็นไปได้** โดยเริ่มจากการหา 1^{st} -Motif, 2^{nd}-Motif, 3^{rd}-Motif, … ตามลำดับ โดยที่ k^{th}-Motif จะต้องไม่ซ้อนทับกับ i^{th}-Motif โดยที่ $1 \leq i < k$ ตัวอย่างเช่น 2^{nd}-Motif จะไม่ซ้อนทับกับ 1^{st}-Motif ที่ถูกค้นพบก่อน ซึ่งแปลความโดยสรุป คือ k^{th}-Motif ที่ค้นพบทั้งหมดจะต้องไม่ซ้อนทับกัน โดยผลลัพธ์ของวิธีการนี้แสดงไว้ ดังภาพที่ 3.5

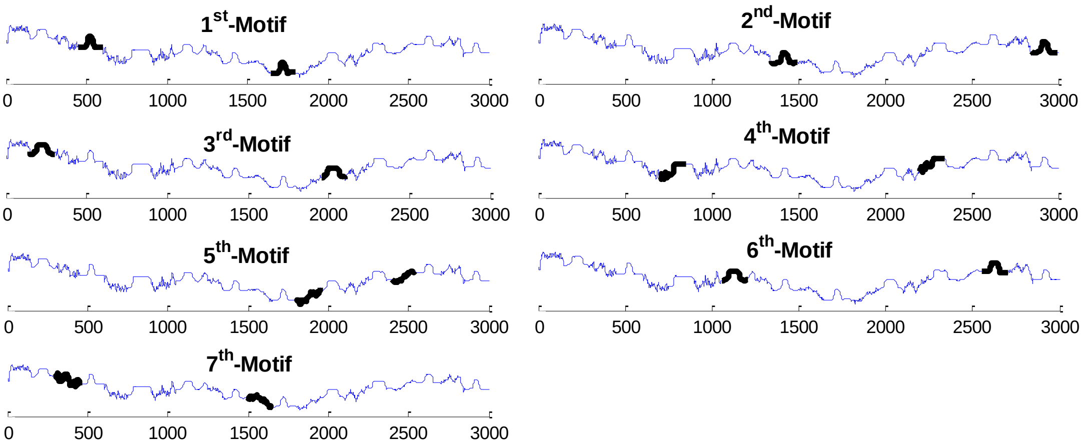


ภาพที่ 3.5 แสดงโมทีฟที่ค้นพบที่ความยาว 150 จุดข้อมูล ซึ่งแสดงตั้งแต่ 1st-Motif ถึง 10th-Motif โดยวิธีการค้นหาโมทีฟทั้งหมดที่เป็นไปได้

วิธีการค้นหาโมทีฟทั้งหมดที่เป็นไปได้ในแต่ละความยาวโมทีฟนี้ต้องใช้คำนวณหาโมทีฟทั้งสิ้นเป็นจำนวนมากที่สุดโดยคำนวณจาก $\binom{n}{r}=\frac{n!}{r!(n-r)!}$ โดย n คือ จำนวนของลำดับย่อยทั้งหมดที่เป็นไปได้ในข้อมูลอนุกรมเวลา T จะสามารถหาค่าของ n ได้จาก $n=|T|-w+1$ โดย w คือ ความยาวโมทีฟ ในที่นี้ T ตามภาพที่ 3.5 มีขนาดความยาว 3000 จุดข้อมูล และ w=150 ดังนั้น $n=3000-150+1=2851$ จากนั้น ค่า r คือ จำนวนลำดับย่อยที่จะเลือกออกมาเป็นคู่ของโมทีฟ ค่า r จึงมีค่าเท่ากับ 2 เพื่อคำนวณว่าถ้ามีจำนวนลำดับย่อยทั้งหมด 2851 จะสามารถเลือกออกมาเป็นคู่โมทีฟได้ทั้งหมดกี่แบบ ดังนั้น จำนวนของโมทีฟที่ต้องทำการค้นหาทั้งหมดที่ความยาวโมทีฟ w=150 คือ $\binom{2851}{2}=\frac{2851!}{2!(2851-2)!}$ ซึ่งการคำนวณค่านี้เป็นจำนวนที่เป็นขอบเขตบนของจำนวนโมทีฟที่เป็นไปได้ทั้งหมดในแต่ละความยาวโมทีฟซึ่งมีจำนวนมหาศาล ปัญหาที่เกิดขึ้นจากการค้นหาโมทีฟที่เป็นไปได้ทั้งหมดนี้จึงเป็นปัญหาทางด้านประสิทธิภาพในการประมวลผล ยกตัวอย่างกรณีหนึ่ง เมื่อเริ่มการค้นหาโมทีฟที่ความยาวของโมทีฟที่ค่าน้อย ๆ และชุดข้อมูลทดลองมีขนาดใหญ่ ตัวอย่างจากข้อมูลทดลองเช่น $|T|$=7000 ซึ่งเป็นขนาดโดยประมาณของข้อมูลทดลองส่วนใหญ่, เมื่ออัลกอริทึมทำการหาโมทีฟทั้งหมดที่ความยาว w=7, และคำนวณค่า $n=7000-7+1=6994$ จะมีจำนวนโมทีฟทั้งหมดที่เป็นไปได้ที่เป็นขอบเขตบนที่ต้องนำมาทำการ

คำนวณหาทั้งสิ้น $\binom{6994}{2} = \frac{6994!}{2!(6994-2)!}$ โมทีฟ ซึ่งเป็นจำนวนโมทีฟที่เกิดขึ้นที่ความยาวเดียวด้วยจำนวนโมทีฟที่มหาศาลนี้ เมื่อต้องค้นหาทุก ๆ k^{th}-Motif ที่เป็นไปได้ในทุกความยาวโมทีฟ ทำให้เป็นปัญหามากกับการใช้เวลาในการประมวลผลโดย Big-O ของวิธีการนี้ คำนวณจาก $m\binom{n}{2}n^2$ เมื่อ m คือ จำนวนค่าความยาวทั้งหมดที่เป็นไปได้ในข้อมูลอนุกรมเวลา T และ n คือจำนวนลำดับย่อยในข้อมูลอนุกรมเวลา T โดยรายละเอียดแล้ว n^2 คือเวลาที่ใช้ในการหาโมทีฟ 1 โมทีฟ $\binom{n}{2}$ คือ จำนวน k^{th}-Motif ทั้งหมดที่เป็นไปได้ในแต่ละความยาวโมทีฟและ m คือ จำนวนค่าของความยาวที่เป็นไปได้ทั้งหมดในข้อมูลอนุกรมเวลา T ซึ่งจะได้ค่าของ Big-O เป็น $O(mn^4)$ โดยเวลาที่ใช้ในการประมวลผลนี้ยังไม่รวมถึงขั้นตอนถัด ๆ ไปของอัลกอริทึมที่มีทั้งขั้นตอนการจัดกลุ่มและฟังก์ชันการให้คะแนน ซึ่งจากการทำการพัฒนาและทดลองแล้วไม่สามารถที่จะทำได้ในระยะเวลาที่จำกัด ผู้วิจัยจึงจำกัดการหาโมทีฟโดยการใช้แนวทางในวิธีที่ 2 ในการหาโมทีฟ

2. **ค้นหาโมทีฟแบบลดจำนวน** โดยเริ่มต้นจากการหา 1^{st}-Motif, 2^{nd}-Motif, 3^{rd}-Motif, … ตามลำดับ เช่นเดียวกับวิธีแรก แต่แตกต่างกันที่คู่ของลำดับย่อยของแต่ละ k^{th}-Motif จะต้องเกิดในตำแหน่งที่ไม่ซ้อนทับกันทั้งหมด ซึ่งในวิธีแรก k^{th}-Motif จะมีลำดับย่อย 1 ลำดับที่สามารถซ้อนทับกันได้ โดยผลลัพธ์ของวิธีการนี้แสดงไว้ ดังภาพที่ 3.6



ภาพที่ 3.6 แสดงโมทีฟที่ค้นพบที่ความยาว 150 จุดข้อมูล ซึ่งแสดงตั้งแต่ 1^{st}-Motif ถึง 7^{th}-Motif

วิธีการค้นหาโมทีฟแบบลดจำนวนนี้ต้องคำนวณหา k^{th}-Motif โดยมีจำนวนโมทีฟที่เป็นขอบเขตบนสูงสุด $\left\lfloor \frac{|T|}{2w} \right\rfloor$ ถ้าขนาดของข้อมูลอนุกรมเวลา T มีขนาด 3000 ที่ความยาวโมทีฟ

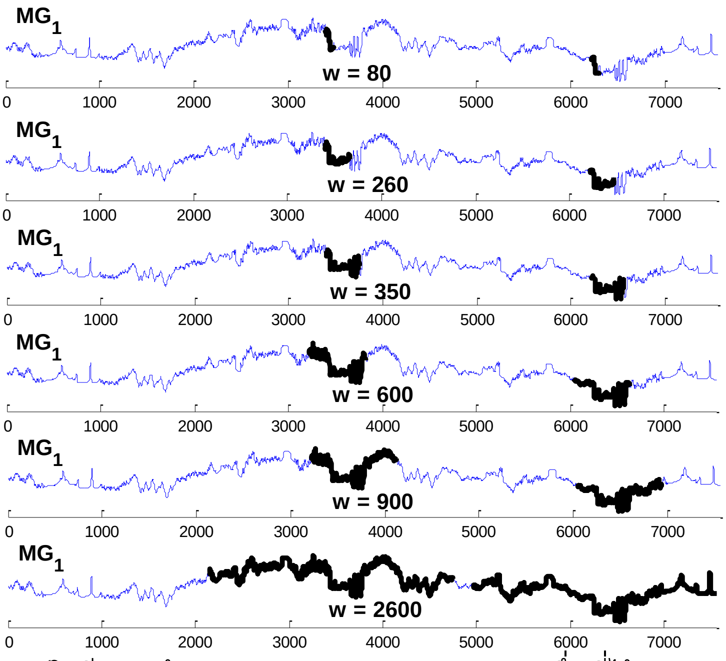
w=150 จะมีจำนวนโมทีฟที่เป็นขอบเขตบนสูงสุด $\left\lfloor \frac{3000}{2(150)} \right\rfloor = 10$ โดยมีความหมายว่าจะสามารถหาคู่ของลำดับย่อยจำนวนสูงสุดได้กี่คู่ลำดับบนข้อมูลอนุกรมเวลาขนาด $|T|$ ดังนั้น จึงต้องนำขนาดของข้อมูลอนุกรมเวลาหารด้วยความยาวของลำดับย่อยจำนวน 2 ลำดับจึงนำความยาวคูณด้วย 2 ซึ่งเมื่อทำการหาโมทีฟตามการทดลองแล้วได้จำนวนทั้งสิ้น 7 โมทีฟ ดังตัวอย่างแสดงในภาพที่ 3.6 จากแนวทางนี้สามารถลดเวลาในการประมวลผลลงเหลือ $O(mn^2)$ และเมื่อลองเปรียบเทียบโมทีฟที่ได้ในวิธีที่ 2 กับวิธีการแรก ดังภาพที่ 3.5 โมทีฟที่ค้นพบในวิธีการแรกที่ 1st-Motif, 2nd-Motif, 3rd-Motif, 4th-Motif, 6th-Motif และ 7th-Motif จะเป็นรูปแบบ GunPoint ที่มีความคล้ายกันจำนวน 4 ลำดับที่ทำการฝังตัวลงไป โดยโมทีฟทั้ง 6 โมทีฟที่ถูกค้นพบนี้จะสลับคู่ไปมาระหว่างกันอยู่ เมื่อเปรียบเทียบกับโมทีฟที่ได้ในวิธีการที่ 2 ที่ 1st-Motif และ 2nd-Motif จะเป็นรูปแบบ GunPoint ที่มีความคล้ายกันจำนวน 4 ลำดับที่ทำการฝังตัวลงไปเช่นกัน ดังนั้น ลำดับทั้งหมดที่ทำการฝังตัวลงไปก็ยังสามารถถูกค้นพบออกมาในวิธีการที่ 2 นี้ อย่างไรก็ตาม ถ้าหากฝังตัวรูปแบบของ GunPoint ที่คล้ายกันลงไปทั้งสิ้น 3 ลำดับหรือจำนวนลำดับที่ทำการฝังลงไปเป็นจำนวนคี่ วิธีการที่ 2 นี้จะสามารถค้นพบลำดับได้เพียงจำนวนคู่เท่านั้น จะมีอยู่ 1 ลำดับที่ไม่ถูกค้นพบออกมา ซึ่งเป็นประเด็นหนึ่งที่เกิดขึ้นจากการใช้วิธีที่ 2 เพื่อลดจำนวนการค้นพบโมทีฟเพื่อลดเวลาการประมวลผลลง ทั้งนี้เนื่องจากหน้าที่ของการค้นพบโมทีฟนั้นเป็นการทำงานส่วนย่อยของกระบวนการหลักของการทำเหมืองข้อมูลพวกการจัดกลุ่มข้อมูล การแยกประเภทข้อมูล ซึ่งการหาโมทีฟนั้นรับหน้าที่ย่อยเพียงเพื่อค้นหารูปแบบที่น่าสนใจออกมาจากข้อมูลอนุกรมเวลา การค้นพบรูปแบบที่น่าสนใจออกมาได้จึงถือว่าเพียงพอกับหน้าที่ของการค้นพบโมทีฟ

โมทีฟที่เป็นผลลัพธ์ทั้งหมดจากขั้นตอนนี้จะนำเข้าสู่การแบ่งกลุ่ม ดังจะกล่าวในหัวข้อถัดไปเรื่องการแบ่งกลุ่มโมทีฟ

3.3 **การแบ่งกลุ่มโมทีฟ**

เมื่อได้ผลลัพธ์โมทีฟมาจากขั้นตอนแรกในการค้นหาโมทีฟ ขั้นตอนการแบ่งกลุ่มโมทีฟนี้มีวัตถุประสงค์เพื่อรวมกลุ่มของโมทีฟที่มีตำแหน่งการค้นพบใกล้เคียงกันและความยาวใกล้เคียงกันให้อยู่ด้วยกัน เพื่อนำไปสู่การลดจำนวนการซ้ำซ้อนของโมทีฟที่มีความคล้ายกันในขั้นตอนการเลือกตัวแทนกลุ่มโมทีฟ ผลลัพธ์โมทีฟทั้งหมดที่ได้จึงนำมาแบ่งกลุ่มโดยใช้ 2 เกณฑ์ ดังนี้

3.3.1 **เกณฑ์การซ้อนทับกันของโมทีฟ** โมทีฟที่ซ้อนทับกันจะถูกจัดให้อยู่ในกลุ่มเดียวกัน โดยผลของการจัดกลุ่มด้วยเกณฑ์การซ้อนทับกันนี้ จะรวมโมทีฟทั้งหมดที่อยู่ในตำแหน่งใกล้เคียงกันไว้ด้วยกันตั้งแต่โมทีฟที่มีขนาดสั้นมากไปจนถึงโมทีฟที่มีความยาวมาก ในที่นี้ขอยกตัวอย่างของข้อมูลอนุกรมเวลาอีกชุดหนึ่งที่เห็นภาพชัดเจนในการทดลองดังภาพที่ 3.7



ภาพที่ 3.7 ผลลัพธ์โมทีฟของข้อมูลอนุกรมเวลาชุดการทดลองหนึ่ง ที่ได้จากการจัดกลุ่มโดยใช้เกณฑ์การซ้อนทับกันของโมทีฟ โดย MG_1 คือ กลุ่มโมทีฟที่ 1 ซึ่งประกอบด้วยโมทีฟที่ความยาว 80, 260, 350, 600, 900 และ 2600

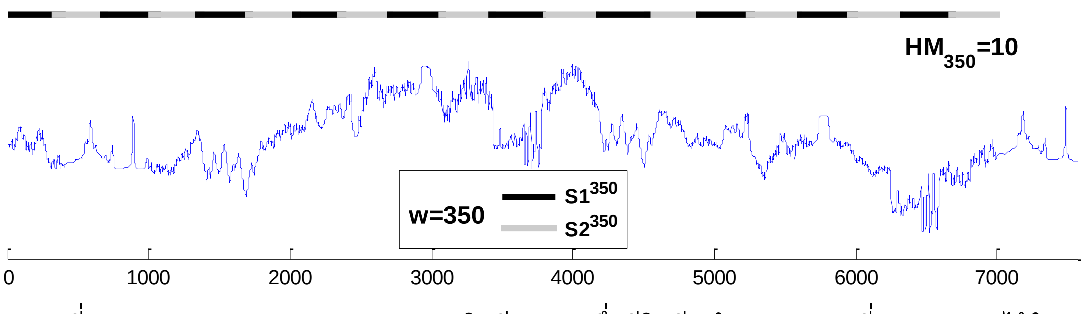
เนื่องจากกลุ่มที่ได้จากการใช้เกณฑ์การซ้อนทับกันของโมทีฟเพียงอย่างเดียวจะทำให้โมทีฟที่มีความยาวแตกต่างกันมากมารวมอยู่ในกลุ่มเดียวกัน โดยโมทีฟที่มีความยาวมากจะซ้อนทับกับรูปแบบที่น่าสนใจหลายตำแหน่งซึ่งไม่ควรเกิดขึ้น และโมทีฟที่มีความยาวน้อยจะเข้ามารวมอยู่ในกลุ่มด้วย ซึ่งโมทีฟขนาดเล็กเหล่านี้ส่งผลกระทบกับขั้นตอนของฟังก์ชันการให้คะแนน "โมทีฟที่ดี" เนื่องจากตำแหน่งการค้นพบโมทีฟขนาดเล็กในแต่ละลำดับ k^{th}-Motif นี้ จะสลับตำแหน่งไปมาไม่คงที่ ฟังก์ชันการให้คะแนนซึ่งใช้ค่าของ k ใน k^{th}-Motif เพื่อคำนวณคะแนนจึงได้รับผลกระทบ ถ้า 1^{st}-Motif (คะแนนมากที่สุดเนื่องจากเป็นโมทีฟที่คู่ลำดับย่อยคล้ายกันมากที่สุด) ของโมทีฟขนาดเล็กถูกค้นพบในตำแหน่งที่ไม่ใช่รูปแบบที่น่าสนใจจะทำให้โมทีฟในตำแหน่งนั้นมีคะแนนมากขึ้นมาทันที ซึ่งส่งผลกระทบต่อการจัดอันดับของโมทีฟ ดังนั้น ผู้วิจัยจึงทำการแบ่งกลุ่มโมทีฟโดยใช้เกณฑ์ขอบเขตบนของจำนวนโมทีฟเข้าร่วมด้วยเพื่อแบ่งโมทีฟที่ซ้อนทับกันและค่าของขอบเขตบนของจำนวนโมทีฟเท่ากันให้อยู่ในกลุ่มเดียวกัน ซึ่งสามารถจำกัดการ

รวมกลุ่มโมทีฟของโมทีฟที่มีความยาวมากและโมทีฟที่มีความยาวน้อยได้ ดังจะกล่าวในหัวข้อถัดไป

3.3.2 **เกณฑ์ขอบเขตบนของจำนวนโมทีฟ** โมทีฟในแต่ละความยาวจะมีค่าขอบเขตบนของจำนวนโมทีฟ โดยการคำนวณหาจำนวนโมทีฟสูงสุดในแต่ละความยาวที่สามารถหาได้ทั้งหมดในข้อมูลอนุกรมเวลาชุดหนึ่ง ดังคำจำกัดความ

ขอบเขตบนของจำนวนโมทีฟ เมื่อมีข้อมูลอนุกรมเวลา T ขนาด n จุดข้อมูล โมทีฟความยาว i คือ M_i และโมทีฟความยาว j คือ M_j จะมีค่าขอบเขตบนของจำนวนโมทีฟเท่ากัน ก็ต่อเมื่อ $\left\lfloor \frac{n}{2i} \right\rfloor = \left\lfloor \frac{n}{2j} \right\rfloor$ เมื่อจำนวนโมทีฟสูงสุดที่สามารถหาได้ของความยาวโมทีฟ i (HM_i) มีค่าเท่ากับ $\left\lfloor \frac{n}{2i} \right\rfloor$ และจำนวนโมทีฟสูงสุดที่สามารถหาได้ของโมทีฟความยาว j (HM_j) มีค่าเท่ากับ $\left\lfloor \frac{n}{2j} \right\rfloor$

ค่าจำนวนโมทีฟสูงสุดที่สามารถหาได้ของแต่ละความยาวโมทีฟ (HM_w) คือ ค่าขอบเขตบนสูงสุดที่โมทีฟความยาวหนึ่งจะสามารถถูกค้นพบบนข้อมูลอนุกรมเวลาขนาดความยาวหนึ่งได้เป็นจำนวนกี่โมทีฟ สามารถอธิบายได้ดังภาพที่ 3.8

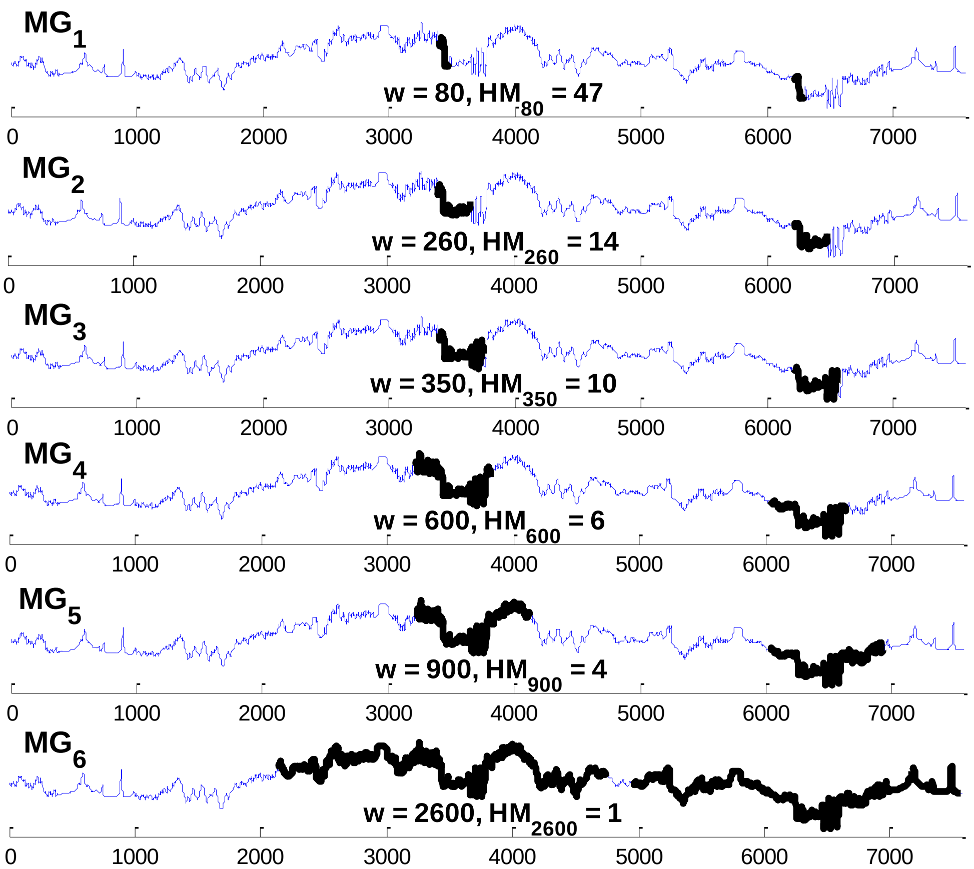


ภาพที่ 3.8 ค่าของ HM_{350} ของความยาวโมทีฟ 350 ซึ่งมีโมทีฟจำนวนสูงสุดที่สามารถหาได้ในข้อมูลอนุกรมเวลาขนาด 7566 จุดข้อมูล จำนวน 10 โมทีฟ โดย $S1^{350}$ และ $S2^{350}$ คือความยาวโมทีฟขนาด 350 ซึ่งใช้แทนคู่ลำดับย่อยของโมทีฟ แสดงด้วยเส้นสีเข้มและเส้นสีจาง

จากภาพที่ 3.8 โมทีฟที่ความยาว 350 จะสามารถมีจำนวนโมทีฟที่สามารถค้นพบได้ทั้งสิ้น 10 โมทีฟ ซึ่งเป็นเช่นเดียวกับ โมทีฟที่ความยาว 351 ไปจนถึงโมทีฟที่ความยาว 378 ซึ่ง

ถ้าคำนวณโดยใช้ $\left\lfloor \dfrac{n}{2w} \right\rfloor$ จะได้ค่าเท่ากับ 10 เสมอ จนกระทั่งความยาวโมทีฟเปลี่ยนเป็น 379 จะได้ค่า HM_{379}=9 ดังนั้น ถ้าโมทีฟที่ความยาว 350 ถึง 378 เกิดการซ้อนทับกันตามเกณฑ์การซ้อนทับกันของโมทีฟแล้ว โมทีฟที่ความยาวในช่วง 350 ถึง 378 จะถูกจัดให้อยู่ในกลุ่มเดียวกัน

เมื่อแบ่งโมทีฟตามเกณฑ์ขอบเขตบนของจำนวนโมทีฟแล้วจะได้ผลลัพธ์ดังภาพที่ 3.9



ภาพที่ 3.9 ผลลัพธ์โมทีฟของข้อมูลอนุกรมเวลา หลังจากการจัดกลุ่มโดยใช้เกณฑ์การซ้อนทับกันของโมทีฟและเกณฑ์ขอบเขตบนของจำนวนโมทีฟ โดยโมทีฟถูกแบ่งออกเป็น 6 กลุ่ม คือ MG_1 ไปจนถึง MG_6 และ HM_w คือ ค่าขอบเขตบนของจำนวนโมทีฟสูงสุดที่สามารถค้นหาได้ในแต่ละความยาว

สังเกตว่าโมทีฟในแต่ละความยาวดังภาพที่ 3.9 ถูกแยกออกจากกันโดยสิ้นเชิง เนื่องจากไม่มีความคล้ายกันตามเกณฑ์ขอบเขตบนของจำนวนโมทีฟ โมทีฟที่มีความยาวน้อยจะถูกคัดออกไปในขั้นตอนนี้ด้วย เนื่องจากค่า HM_w ของโมทีฟขนาดเล็กจะไม่เท่ากับโมทีฟที่ความยาวอื่นเลย ยกตัวอย่างเช่น ถ้าโมทีฟที่มีขนาดความยาว 2, 3, 4, และ 5 จุดข้อมูล บนข้อมูลอนุกรม

เวลา T ที่มีขนาด 3000 จุดข้อมูล จะสามารถคำนวณหาค่า HM_w ของโมทีฟที่ความยาวเหล่านี้ ได้เท่ากับ $\left\lfloor \frac{3000}{2(2)} \right\rfloor = 750, \left\lfloor \frac{3000}{2(3)} \right\rfloor = 500, \left\lfloor \frac{3000}{2(4)} \right\rfloor = 375,$ และ $\left\lfloor \frac{3000}{2(5)} \right\rfloor = 300$ ตามลำดับ โมทีฟที่มีขนาดเล็กเหล่านี้จึงถูกคัดออกไปในขั้นตอนการแบ่งกลุ่ม เนื่องด้วยค่าของ HM_w ของโมทีฟไม่เท่ากับโมทีฟที่ความยาวอื่นเลย

ผลลัพธ์หลังจากการจัดกลุ่มโมทีฟให้อยู่ในเกณฑ์การซ้อนทับกันและเกณฑ์ขอบเขตบนของจำนวนโมทีฟนี้ ทำให้ได้กลุ่มของโมทีฟที่มีความใกล้เคียงกันทั้งตำแหน่งการเกิดและประมาณได้ว่ามีความใกล้เคียงกันทางความยาวด้วยเมื่อขอบเขตบนของจำนวนโมทีฟมีค่าเท่ากัน ซึ่งสามารถลดความซ้ำซ้อนของโมทีฟที่มีลักษณะใกล้เคียงกันนี้ โดยการนำเข้าสู่ขั้นตอนถัดไป คือ การหาตัวแทนของกลุ่มโมทีฟ

3.4 การหาตัวแทนของกลุ่มโมทีฟ

เมื่อโมทีฟแต่ละกลุ่มมีความใกล้เคียงกันทั้งตำแหน่งการค้นพบและโมทีฟมีความยาวที่ใกล้เคียงกัน เซตของโมทีฟทั้งหมดจึงสามารถลดรูปลงโดยการเลือกตัวแทนของโมทีฟจากกลุ่มโมทีฟแต่ละกลุ่ม ในขั้นตอนนี้อัลกอริทึมจะทำการหาตัวแทนโมทีฟจากกลุ่มแต่ละกลุ่ม โดยการคำนวณค่าของระยะทางยุคลิดโดยปรับให้อยู่ในอัตราส่วนเดียวกันเนื่องจากโดยการคำนวณแล้วความยาวโมทีฟยิ่งมากขึ้นแนวโน้มของระยะทางยุคลิดจะเพิ่มขึ้นด้วยและโมทีฟภายในกลุ่มเดียวกันจะมีความยาวที่แตกต่างกัน ดังนั้น การจะเปรียบเทียบความคล้ายของโมทีฟภายในกลุ่มจึงต้องปรับอัตราส่วนของระยะทางให้อยู่ในอัตราส่วนเดียวกันก่อน ผู้วิจัยจึงทำการปรับอัตราส่วนของระยะทางยุคลิดของแต่ละคู่ลำดับย่อยของโมทีฟภายในกลุ่มซึ่งมีความยาวที่แตกต่างกัน ตามสมการ

$$NDist_W = \frac{\mathrm{EUC}(S_i^{w}\ , S_j^{w})}{w} \tag{3.1}$$

$NDist_w$ คือ ค่าระยะทางยุคลิดในอัตราส่วนเดียวกันของโมทีฟที่ความยาว w โดย $\mathrm{EUC}(S_i^w, S_j^w)$ คือ ระยะทางยุคลิดของคู่ลำดับย่อย S_i^w, S_j^w ที่ความยาว w โดย i และ j คือตำแหน่งเริ่มต้นของลำดับย่อย

ค่าระยะทางยุคลิดที่อยู่ในอัตราส่วนเดียวกัน (Normalized Euclidean Distance - *NDist*) จะเป็นค่าที่ใช้ในการเปรียบเทียบความดีระหว่างโมทีฟภายในกลุ่ม โดยโมทีฟที่มีค่าระยะทางยุคลิดที่อยู่ในอัตราส่วนเดียวกันน้อยที่สุดจะถูกเลือกให้เป็นตัวแทนของกลุ่มโมทีฟกลุ่ม

นั้น ตัวอย่างเช่น โมทีฟในกลุ่มที่ 1 มีโมทีฟที่ความยาว 350, 351, 352, 353, และ 354 โดยการคำนวณค่าระยะทางยุคลิดที่อยู่ในอัตราส่วนเดียวกันของแต่ละโมทีฟได้ผลลัพธ์เป็น 1.59, 1.63, 1.41, 1.89, และ 2.01 ตามลำดับ ดังนั้น โมทีฟที่ความยาว 352 ที่มีค่าระยะทาง 1.41 ซึ่งน้อยที่สุดจะเป็นตัวแทนกลุ่มของโมทีฟกลุ่มที่ 1 นี้

3.5 วิธีการคำนวณคะแนน "โมทีฟที่ดี"

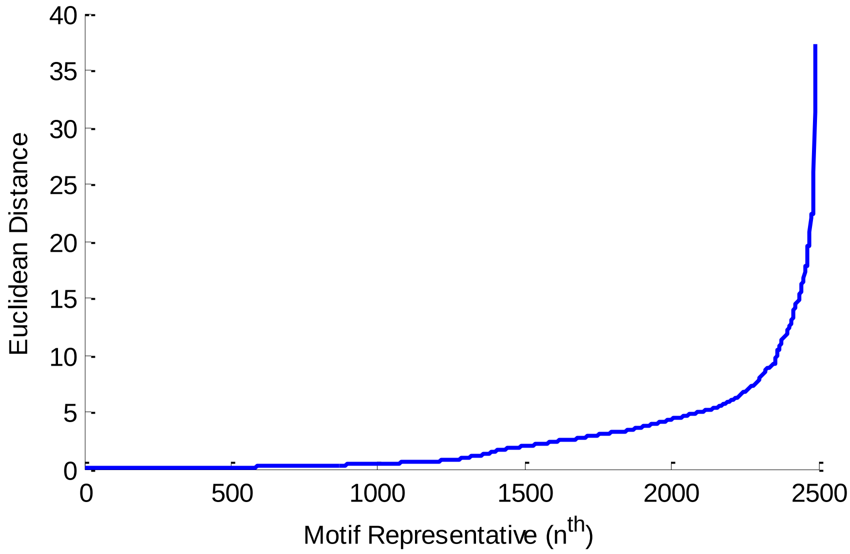
หลังจากโมทีฟถูกลดจำนวนลง โมทีฟที่ถูกเลือกมาเป็นตัวแทนทั้งหมดนี้ จะนำมาเข้าสู่กระบวนการประมวลผลเพื่อหา "โมทีฟที่ดี" โดยมีขั้นตอนดังต่อไปนี้

3.5.1 **การคัดออกตัวแทนโมทีฟที่มีความยาวมาก** เนื่องจากโมทีฟที่มีความยาวมากเกินนั้นจะครอบคลุมตำแหน่งเป็นบริเวณกว้างทำให้รูปแบบที่น่าสนใจในหลายตำแหน่งถูกรวมเข้าไปอยู่ในโมทีฟเดียวกัน ซึ่งไม่ควรเกิดขึ้น ดังตัวอย่างในภาพที่ 3.9 ดังนั้น ตัวแทนโมทีฟของกลุ่มโมทีฟกลุ่มที่ 6 ที่ความยาว 2600 จุดข้อมูล มีความยาวมากจนครอบคลุมเกินครึ่งหนึ่งของข้อมูลอนุกรมเวลาทั้งหมด ดังนั้น ในขั้นตอนนี้จึงทำการคัดโมทีฟที่มีความยาวมากเกินออก โดยพิจารณาจากระยะทางยุคลิดของตัวแทนโมทีฟ ซึ่งหมายถึงค่า *R* ตามคำจำกัดความของ "โมทีฟที่ดี"

> **โมทีฟที่ดี** (Best Motifs) คือ คู่ของลำดับย่อย (S_i^w, S_j^w) ที่มีรูปร่างคล้ายกันที่ยาวที่สุด ที่ถูกค้นพบที่ตำแหน่งใดตำแหน่งหนึ่ง โดยที่ $EUC(S_i^w, S_j^w) \leq R$ และ *R* คือระยะทางยุคลิดที่มากที่สุดที่ใช้ในการพิจารณาความเป็นโมทีฟที่ดี

ดังนั้น อัลกอริทึมต้องทำการหาค่า *R* ที่เหมาะสมกับทุกข้อมูลอนุกรมเวลา เพื่อเป็นค่าที่จะใช้พิจารณาว่าโมทีฟที่มีความคล้ายกันน้อย คือ มีระยะทางยุคลิดสูงกว่าค่า *R* ต้องทำการคัดออก โดยค่า *R* นี้ อัลกอริทึมจะค้นหาจาก ตัวแทนโมทีฟทั้งหมดที่ได้จากการเลือกตัวแทนกลุ่มโมทีฟ ตัวแทนโมทีฟนี้จะนำมาเรียงลำดับจากโมทีฟที่มีค่าระยะทางยุคลิดของคู่ลำดับย่อยจากน้อยไปมาก แล้วนำค่าระยะทางยุคลิดของโมทีฟที่อยู่ในตำแหน่งมัธยฐานมาใช้เป็นค่า *R* ในการพิจารณาเพื่อคัดโมทีฟที่มีระยะทางยุคลิดมากเกินค่านี้ออกไป ค่าของระยะทางยุคลิดในตำแหน่งมัธยฐานนี้ถูกเลือกจากการพิจารณาตัวแทนโมทีฟของข้อมูลทดลองทั้งหมด เนื่องจากข้อมูลอนุกรมเวลาแต่ละข้อมูลอาจมีรูปแบบที่น่าสนใจฝังตัวอยู่แตกต่างกันออกไปซึ่งในความเป็นจริงแล้วไม่สามารถรู้ได้ว่าจำนวนของรูปแบบที่น่าสนใจเป็นเท่าใดและมีขนาดความยาวที่แน่นอนเป็นเท่าใด สิ่งที่นำมาพิจารณาได้ คือ ขนาดของข้อมูลอนุกรมเวลาแต่ละข้อมูลและจำนวนของตัวแทน

โมทีฟที่ได้ทั้งหมด ซึ่งจำนวนของตัวแทนโมทีฟนั้นจะมีแนวโน้มในการเพิ่มจำนวนมากขึ้นเมื่อข้อมูลอนุกรมเวลามีขนาดความยาวมากขึ้น โดยตัวแทนโมทีฟจะมีขนาดตั้งแต่ขนาดเล็กไปจนถึงขนาดใหญ่ดังแสดงในภาพที่ 3.9 ซึ่งเมื่อทำการเรียงลำดับจากโมทีฟที่มีระยะทางยุคลิดจากน้อยที่สุดไปมากที่สุดจะได้ผลดังภาพที่ 3.10

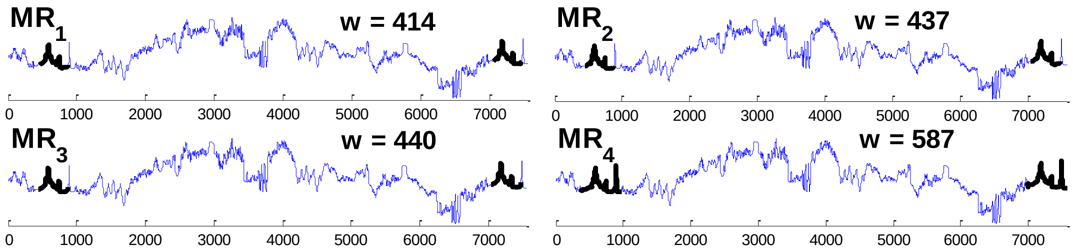


ภาพที่ 3.10 กราฟเส้นแสดงค่าของระยะทางยุคลิดของตัวแทนโมทีฟที่จัดอันดับจากน้อยไปมากของข้อมูลทดลองชุดที่ 1 ที่ขนาดความยาว 7566 จุดข้อมูล

จากกราฟที่แสดงดังภาพที่ 3.10 ตำแหน่งของตัวแทนโมทีฟโดยประมาณที่ตำแหน่งมัธยฐานจะเป็นจุดที่ระยะทางยุคลิดเริ่มมีแนวโน้มเพิ่มสูงขึ้น ดังนั้น ค่าของระยะทางยุคลิดที่ตำแหน่งมัธยฐานจึงถูกเลือกให้เป็นค่า *R* เพื่อพิจารณา “โมทีฟที่ดี” และค่า *R* นี้จะแตกต่างกันออกไปตามข้อมูลอนุกรมเวลา ดังที่กล่าวเบื้องต้นว่า จำนวนของตัวแทนโมทีฟจะมีแนวโน้มเพิ่มสูงขึ้นตามขนาดความยาวของข้อมูลอนุกรมเวลาที่เพิ่มมากขึ้น ดังนั้น ค่าของระยะทางยุคลิดที่ตำแหน่งมัธยฐานจึงเป็นค่าที่ถูกปรับให้เหมาะสมตามข้อมูลอนุกรมเวลาแต่ละชุดข้อมูล ในท้ายที่สุดตัวแทนโมทีฟที่มีระยะทางยุคลิดเกินค่า *R* จะถูกคัดออกจากการพิจารณา จากการทดลองในขั้นตอนนี้ได้ทดลองกับค่ากลางหลาย ๆ ค่า คือ ค่าเฉลี่ย มัธยฐาน และฐานนิยม ซึ่งให้ผลที่ใกล้เคียงกันแต่เนื่องจากค่าของมัธยฐานนั้นให้ผลที่ดีกว่าในงานวิจัยนี้จึงเลือกให้เป็นค่า *R* จากนั้นตัวแทนโมทีฟที่เหลืออยู่ทั้งหมดจะนำเข้าสู่ขั้นตอนการรวมกลุ่มตัวแทนโมทีฟ

3.5.2 **การรวมกลุ่มตัวแทนโมทีฟ** หลังจากการคัดออกโมทีฟที่มีระยะทางยุคลิดเกินค่า *R* ในขั้นตอนที่แล้ว ตัวแทนโมทีฟที่เหลือจะนำมารวมกลุ่มเพื่อทำการคำนวณคะแนน โดยการรวมกลุ่มตัวแทนโมทีฟนี้จะใช้เกณฑ์การซ้อนทับกันของโมทีฟในการรวมกลุ่มเท่านั้น ซึ่ง

แตกต่างกับการแบ่งกลุ่มโมทีฟในหัวข้อ 3.2 เนื่องจากอัลกอริทึมได้ทำการคัดโมทีฟที่มีความยาวมากและโมทีฟที่มีความยาวน้อยออกไปแล้ว ในขั้นตอนนี้จึงเป็นการรวมกลุ่มเพื่อหาโมทีฟที่ยาวที่สุดในแต่ละตำแหน่งเพื่อให้ได้โมทีฟที่ครอบคลุมรูปแบบที่น่าสนใจมากที่สุด โดยมีตัวอย่างของตัวแทนโมทีฟที่จะถูกรวมอยู่ในกลุ่มเดียวกัน ดังภาพที่ 3.11



ภาพที่ 3.11 แสดงตัวแทนโมทีฟ (MR) จำนวน 4 โมทีฟที่ความยาว 414, 437, 440, 587 ที่ซ้อนทับกันและจะรวมอยู่ในกลุ่มเดียวกันตามเกณฑ์การซ้อนทับกันของโมทีฟ โดยมีโมทีฟที่ความยาว 587 เป็นโมทีฟที่ยาวที่สุดในกลุ่ม

เมื่อตัวแทนโมทีฟถูกแบ่งออกเป็นกลุ่ม ๆ ตัวแทนโมทีฟที่มีความยาวมากที่สุดของแต่ละกลุ่มจะถูกเลือกออกมาเป็น "โมทีฟที่ดี" ดังภาพที่ 3.11 โมทีฟที่ความยาว 587 จะเป็น "โมทีฟที่ดี" ของตัวแทนโมทีฟกลุ่มนี้ เนื่องจากโมทีฟที่มีความยาวมากถูกคัดออกไปในขั้นตอนแรก ตัวแทนโมทีฟที่เหลืออยู่เมื่อรวมกลุ่มตามเกณฑ์การซ้อนทับกันแล้ว ตัวแทนโมทีฟที่มีความยาวมากที่สุดจึงเป็นโมทีฟที่ครอบคลุมรูปแบบที่น่าสนใจมากที่สุด ณ ตำแหน่งนั้น ซึ่งเป็นไปตามคำจำกัดความของ "โมทีฟที่ดี" ที่ว่าเป็นโมทีฟที่ยาวที่สุดที่ถูกค้นพบ ณ ตำแหน่งใดตำแหน่งหนึ่งที่มีค่าระยะทางยุคลิดไม่เกินค่า *R* หลังจากที่ได้ผลลัพธ์โมทีฟออกมาเป็นกลุ่มตัวแทนโมทีฟที่มี "โมทีฟที่ดี" ที่มีความยาวมากที่สุดในกลุ่มแล้ว ขั้นตอนถัดไป คือ การจัดอันดับ "โมทีฟที่ดี" ของแต่ละกลุ่มตัวแทนโมทีฟ ว่าโมทีฟที่ดีของกลุ่มตัวแทนโมทีฟกลุ่มใดเป็นโมทีฟที่ดีที่สุดเรียงอันดับออกมาเป็นเซตของคำตอบให้ผู้ใช้นำไปใช้งาน ซึ่งจะกล่าวในลำดับถัดไป

3.5.3 **การคำนวณคะแนนเพื่อจัดอันดับ "โมทีฟที่ดี"**

การคำนวณคะแนนของ "โมทีฟที่ดี" นั้น สามารถคำนวณได้จากความคล้ายกันของสมาชิกภายในกลุ่มตัวแทนโมทีฟ เนื่องจากสมาชิกตัวแทนโมทีฟจะเป็นส่วนหนึ่งของรูปแบบที่น่าสนใจ ณ ตำแหน่งนั้น ซึ่งมีความยาวน้อยกว่า "โมทีฟที่ดี" ของกลุ่ม สมาชิกเหล่านั้นจึงมีลักษณะที่สามารถบ่งบอกความดีของโมทีฟได้ ซึ่งแบ่งออกเป็น 2 ลักษณะ คือ

1. **ความคล้ายกันของแต่ละสมาชิกตัวแทนโมทีฟภายในกลุ่ม** ถ้าแต่ละตัวแทนโมทีฟภายในกลุ่มมีคู่ของลำดับย่อยที่มีความคล้ายกันมากกว่ากลุ่มอื่น “โมทีฟที่ดี” ของกลุ่มตัวแทนโมทีฟนั้นจะดีที่สุด

2. **จำนวนของโมทีฟที่เกิด ณ ตำแหน่งนั้น** ตำแหน่งของข้อมูลอนุกรมเวลาที่มีรูปร่างคล้ายกันมาก จะทำให้จำนวนโมทีฟที่เกิดขึ้น ณ ตำแหน่งนั้น มีจำนวนมากตามไปด้วย

จากลักษณะทั้ง 2 ข้อนี้ จึงสามารถคำนวนคะแนนเพื่อจัดอันดับของ “โมทีฟที่ดี” ได้โดยกำหนดให้ตัวแทนโมทีฟแต่ละตัวแทนมีคะแนนในตัวมันเองโดยขึ้นอยู่กับ k^{th}-Motif ของตัวแทนโมทีฟนั้น โดยความหมายแล้วค่า k ของ k^{th}-Motif จะเป็นค่าที่บ่งบอกความคล้ายกันของคู่ลำดับย่อยในโมทีฟแต่ละโมทีฟ ตัวอย่างเช่น ที่ความยาว w=100, 1^{st}-Motif จะมีคู่ลำดับย่อยที่คล้ายกันมากกว่า 2^{nd}-Motif ดังนั้น อัลกอริทึมจึงกำหนดคะแนนของตัวแทนโมทีฟแต่ละตัวแทนนี้ให้มีค่า 1/k เมื่อแต่ละตัวแทนมีคะแนนของตัวมันเองแล้ว สืบเนื่องจากการบ่งบอกความดีลักษณะที่ 2 ข้างต้น ซึ่งขึ้นกับจำนวนโมทีฟ อัลกอริทึมจึงนำผลรวมของคะแนนของตัวแทนโมทีฟภายในกลุ่มมาเป็นค่าความดีของโมทีฟที่ดี ซึ่งจะทำให้กลุ่มที่มีจำนวนสมาชิกมากมีคะแนนเพิ่มขึ้นมาก โดยสามารถคำนวณคะแนนของแต่ละกลุ่มได้ ตามสมการ

$$\text{Score}_{MRG} = \sum_{i=1}^{r} \frac{1}{k_i} \tag{3.2}$$

ค่า Score_{MRG} คือ คะแนนของกลุ่มตัวแทนโมทีฟ MRG, n คือ จำนวนสมาชิกตัวแทนโมทีฟภายในกลุ่ม, และ k คือ ค่าของ k ใน k^{th}-Motif ของแต่ละตัวแทนโมทีฟภายในกลุ่ม โดยมีตัวอย่างของการคำนวณคะแนน ดังต่อไปนี้

กลุ่มตัวแทนโมทีฟ 2 กลุ่ม คือ {1^{st}-M_{100}, 2^{nd}-M_{200}, 5^{th}-M_{300}} และ {2^{nd}-M_{150}, 2^{nd}-M_{250}, 7^{th}-M_{350}} โดย M_w คือ โมทีฟที่มีความยาว w ดังนั้น คะแนนของกลุ่มตัวแทนโมทีฟทั้ง 2 กลุ่มนี้จะมีค่าเท่ากับ {1+1/2+1/5=1.7} และ {1/2+1/2+1/7=1.14} จากตัวอย่างนี้ กลุ่มที่ 1 มีคะแนนมากกว่ากลุ่มที่ 2 ดังนั้น “โมทีฟที่ดี” ของกลุ่มที่ 1 ซึ่งก็คือ 5^{th}-M_{300} ที่มีความยาวโมทีฟเท่ากับ 300 ซึ่งยาวมากที่สุดในกลุ่ม จะเป็นโมทีฟที่ดีที่สุดของตัวอย่างนี้ ส่วน “โมทีฟที่ดี” ของกลุ่มที่ 2 ซึ่งก็คือ 7^{th}-M_{350} ซึ่งมีความยาว 350 ซึ่งเป็นโมทีฟที่มีความยาวมากที่สุดในกลุ่ม จะเป็น “โมทีฟที่ดี” ที่อยู่ในอันดับที่ 2 โดยได้ผลลัพธ์การจัดอันดับเป็นเซตของ "โมทีฟที่ดี” คือ 5^{th}-M_{100} และ 7^{th}-M_{350} ตามลำดับ ซึ่งจะได้เซตของคำตอบสุดท้ายของอัลกอริทึมเป็น {5^{th}-BM_{300}, 7^{th}-BM_{350}}

เนื้อความโดยสรุปของบทนี้ ได้นำเสนอขั้นตอนของวิธีการหาโมทีฟความยาวแปรผันในข้อมูลอนุกรมเวลา ซึ่งประกอบไปด้วยคำจำกัดความที่ใช้ในงานวิจัย จากนั้นจึงกล่าวถึงขั้นตอนการหาโมทีฟที่ความยาวแตกต่างกันรวมไปถึงการค้นหาโมทีฟโดยใช้แนวคิดของ k^{th}-Time Series Motif จากนั้น จึงอธิบายขั้นตอนของการแบ่งกลุ่มโมทีฟแล้วเลือกตัวแทนกลุ่มเพื่อลดจำนวนโมทีฟลงแล้วนำตัวแทนกลุ่มไปใช้ในขั้นตอนการหา “โมทีฟที่ดี” แล้วจึงทำการคำนวณคะแนนเพื่อจัดอันดับโมทีฟ ในส่วนของการทดลองของวิธีการนี้จะนำเสนอในลำดับถัดไป

บทที่ 4

การทดลองและวิเคราะห์ผลการทดลอง

หลังจากที่ได้อธิบายถึงขั้นตอนของอัลกอริทึมที่นำเสนอในงานวิจัยชิ้นนี้ไปในบทที่แล้ว ในบทนี้จะกล่าวถึงการทดลองทั้งหมดของอัลกอริทึมการค้นพบโมทีฟความยาวแปรผันในข้อมูลอนุกรมเวลา โดยในส่วนเนื้อหานั้นจะเริ่มต้นด้วยการเตรียมข้อมูลทดลองทั้งหมดซึ่งประกอบไปด้วย 16 ชุดข้อมูลทดลอง จากนั้นจึงอธิบายวิธีการวัดผลและประเมินผลการทดลองซึ่งประกอบไปด้วยขั้นตอนการวัดผล 2 ขั้นตอน คือ วัดคุณภาพของโมทีฟที่อัลกอริทึมค้นพบและวัดคุณภาพของฟังก์ชันการให้คะแนนในการจัดอันดับโมทีฟ ถัดไปจึงเป็นการแสดงผลการทดลองทั้งหมดซึ่งในงานวิจัยนี้เปรียบเทียบกับงานวิจัยที่ผ่านมา [11] รวมไปถึงการวิเคราะห์ผลการทดลองและสรุปผลการทดลองเป็นลำดับสุดท้าย

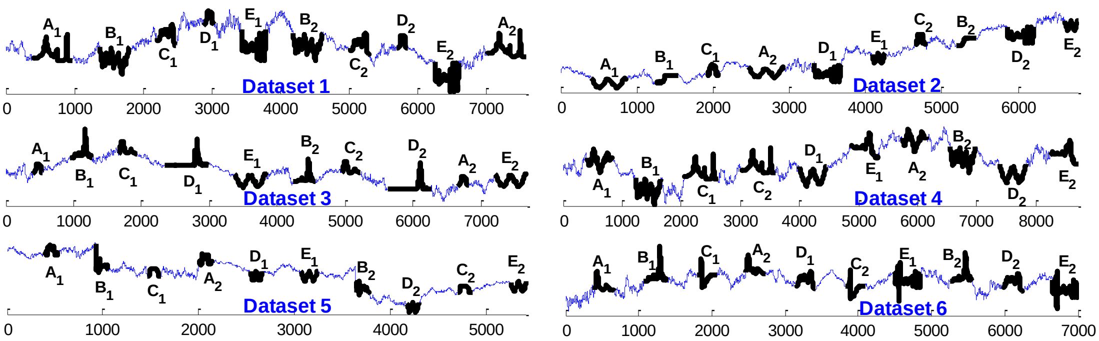
4.1 การเตรียมข้อมูลทดลอง

ข้อมูลทดลองที่ใช้ทั้งหมดเป็นข้อมูลทดลองที่นำมาจากข้อมูลการจัดกลุ่มและการจำแนกประเภทข้อมูลอนุกรมเวลาของ UCR [1] แล้วทำการสร้างชุดข้อมูลทดลองโดยการฝังแต่ละรูปแบบของข้อมูล [1] ลงไประหว่างข้อมูลแบบสุ่ม (Randomwalk Data) ซึ่งรายละเอียดของรูปแบบข้อมูลที่ทำการฝังลงไปในข้อมูลแบบสุ่มนี้ แยกออกเป็น 2 ส่วน ส่วนแรก คือ ชุดข้อมูลทดลองที่ทำการฝังตัวรูปแบบของข้อมูลหลายรูปแบบและส่วนที่สอง คือ ชุดข้อมูลที่ทำการฝังตัวรูปแบบของข้อมูลรูปแบบเดียวซึ่งมีความยาวเดียว

4.1.1 ชุดข้อมูลที่ฝังตัวรูปแบบหลายรูปแบบ ชุดข้อมูลในส่วนแรกนี้ประกอบไปด้วย 6 ชุดข้อมูล โดยชุดข้อมูลหนึ่งจะฝังรูปแบบข้อมูลลงไป 5 รูปแบบข้อมูล ข้อมูลแต่ละรูปแบบจะถูกเลือกออกมา 1 คู่จากลำดับข้อมูลทั้งหมด โดยการเลือกคู่ที่คล้ายกันมากที่สุดมาทำการฝังตัวลงในชุดข้อมูลทดลอง รายละเอียดของรูปแบบทั้งหมดในแต่ละชุดข้อมูลทดลองแสดงไว้ในตารางที่ 4.1 และข้อมูลทดลองในส่วนที่ 1 ทั้ง 6 ชุดข้อมูลนี้ แสดงไว้ดังภาพที่ 4.1

ตารางที่ 4.1 รูปแบบทั้งหมดที่ทำการฝังตัวลงในชุดข้อมูลส่วนที่ 1 จำนวน 6 ชุดข้อมูล

ชุดข้อมูล	รูปแบบ	*w*	ชุดข้อมูล	รูปแบบ	*w*	ชุดข้อมูล	รูปแบบ	*w*
ชุดข้อมูล 1	OliveOil	570	ชุดข้อมูล 2	FISH	463	ชุดข้อมูล 3	FISH	463
	OSULeaf	427		Adiac	176		Lightning2	637
	Coffee	286		Trace	275		50words	270
	GunPoint	150		FaceFour	350		GunPoint	150
	FaceFour	350		CBF	128		Lightning7	319
ชุดข้อมูล 4	OliveOil	570	ชุดข้อมูล 5	Adiac	176	ชุดข้อมูล 6	Trace	275
	OSULeaf	427		SwedishLeaf	128		Coffee	286
	FISH	463		FaceAll	131		50words	270
	Beef	470		GunPoint	150		Lightning7	319
	Yoga	426		CBF	128		FaceFour	350

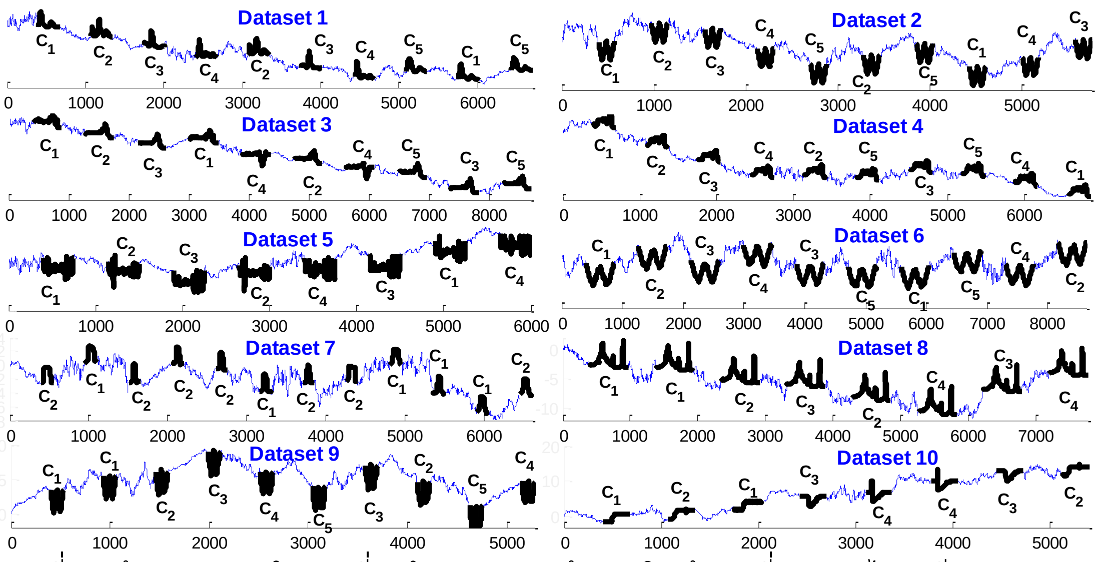


ภาพที่ 4.1 ข้อมูลทดลองในส่วนที่ 1 จำนวน 6 ชุดข้อมูล โดยข้อมูลที่ฝังตัวลงไปแต่ละรูปแบบจะแทนด้วยตัวอักษร A, B, C, D, และ E

4.1.2 **ชุดข้อมูลที่ทำการฝังตัวข้อมูลรูปแบบเดียว** ข้อมูลทดลองในส่วนที่ 2 นี้ประกอบไปด้วยชุดข้อมูลทดลองจำนวน 10 ชุดข้อมูล ซึ่งแต่ละชุดข้อมูลจะฝังตัวรูปแบบของข้อมูลลงไปรูปแบบเดียวจึงมีข้อมูลที่ความยาวเดียว โดยรูปแบบของข้อมูลแต่ละรูปแบบนี้จะแบ่งออกได้เป็นหลายประเภทของข้อมูลเนื่องจากเป็นข้อมูลที่ใช้ในการทำการแยกประเภทข้อมูล (Classification) ดังนั้น ชุดข้อมูลทดลองในส่วนที่ 2 จะทำการคัดเลือกคู่ของข้อมูลที่คล้ายกันมากที่สุดของแต่ละประเภท (Class) มาทำการฝังลงไปในชุดข้อมูล โดยรายละเอียดแสดงไว้ในตารางที่ 4.2 และข้อมูลทดลองในส่วนที่ 2 ทั้ง 9 ชุดข้อมูลนี้ แสดงไว้ดังภาพที่ 4.2

ตารางที่ 4.2 รูปแบบทั้งหมดที่ทำการฝังตัวลงในข้อมูลส่วนที่ 2 จำนวน 10 ชุดข้อมูล

ชุดข้อมูล	**รูปแบบ**	*w*	**จำนวนประเภท**	**ชุดข้อมูล**	**รูปแบบ**	*w*	**จำนวนประเภท**
ชุดข้อมูล 1	50words	270	5	ชุดข้อมูล 2	Adiac	176	5
ชุดข้อมูล 3	Beef	470	5	ชุดข้อมูล 4	Coffee	286	5
ชุดข้อมูล 5	FaceFour	350	4	ชุดข้อมูล 6	FISH	463	5
ชุดข้อมูล 7	GunPoint	150	2	ชุดข้อมูล 8	OliveOil	570	4
ชุดข้อมูล 9	SwedishLeaf	128	5	ชุดข้อมูล10	Trace	275	4



ภาพที่ 4.2 ข้อมูลทดลองในส่วนที่ 2 จำนวน 10 ชุดข้อมูล โดยข้อมูลที่ฝังตัวลงไปจะมี 1 รูปแบบต่อ 1 ชุดข้อมูล แต่ประกอบไปด้วยข้อมูลหลายประเภท (Class) ซึ่งจะแทนประเภทของข้อมูลด้วยตัวอักษร C_i โดย *i* คือ ประเภทของข้อมูล

ชุดข้อมูลทั้งหมดนี้จะนำเข้าสู่การทดลองโดยเปรียบเทียบระหว่างวิธีการของอัลกอริทึมที่นำเสนอ ซึ่งมีชื่อว่า อัลกอริทึมการค้นพบโมทีฟที่ดี (*k*-Best Motif Discovery Algorithm - *k*BMD) กับวิธีการของงานวิจัยที่เกี่ยวข้องซึ่งใช้วิธีการแปลงข้อมูลอนุกรมเวลาให้เป็นข้อมูลวิยุตโดยใช้วิธีการของ SAX [13] ดังนั้น ในการทดลองนี้จะใช้ชื่อของอัลกอทึมที่ทำการเปรียบเทียบนี้ว่า อัลกอริทึมการค้นพบโมทีฟโดยใช้ SAX (Motif Discovery based on SAX - MDS) ลำดับถัดไปจะนำเสนอวิธีการวัดผลและประเมินผลการทดลอง

4.2 วิธีการวัดผลและประเมินผลการทดลอง

วิธีการวัดผลและประเมินผลการทดลองนี้แบ่งออกเป็น 3 ขั้นตอนหลัก คือ การวัดคุณภาพของโมทีฟที่ถูกค้นพบจากอัลกอริทึม ขั้นตอนที่สองจะทำการวัดประสิทธิภาพของฟังก์ชันการให้คะแนนของ *k*BMD อัลกอริทึมและขั้นตอนที่สามเป็นส่วนที่วัดความสามารถในการลดจำนวนคำตอบของโมทีฟที่เป็นไปได้ทั้งหมดเพื่อสนับสนุนการวัดผลในขั้นตอนที่สองว่าผลของการจัดอันดับของฟังก์ชันในการให้คะแนนถึงแม้โมทีฟทั้งหมดที่เป็นไปได้จะมีจำนวนมากแต่ผลลัพธ์ที่ได้จากการจัดอันดับก็ยังคงประสิทธิภาพที่ดีในการค้นพบรูปแบบที่ทำการฝังตัวลงไป

ในงานวิจัยชิ้นนี้ใช้วิธีการวัดคุณภาพของโมทีฟจากการนำวิธีการ Accuracy-on-Detection (AoD) [14] มาใช้งาน แต่ในส่วนของการวัดประสิทธิภาพฟังก์ชันการให้คะแนนได้นำวิธีการ Accuracy-on-Retrieval (AoR) [14] มาใช้งาน ซึ่งทั้งสองวิธีการมีการปรับเปลี่ยนให้เหมาะสมกับงานชิ้นนี้ และในส่วนของความสามารถในการลดจำนวนคำตอบที่เป็นไปได้ จะใช้วิธีการคำนวณหาขอบเขตบนของจำนวนโมทีฟสูงสุดที่เป็นไปได้ในแต่ละข้อมูลอนุกรมเวลาแล้วเปรียบเทียบกับจำนวน "โมทีฟที่ดี" k^{th}-BM_w ในเซตคำตอบของอัลกอริทึมเพื่อคำนวณหาอัตราการลดจำนวนโมทีฟว่าเป็นกี่เปอร์เซนต์ โดยรายละเอียดของวิธีการทั้งสามเป็นดังนี้

4.2.1 Accuracy-on-Detection (AoD) วิธีการนี้ใช้ในการวัดคุณภาพของโมทีฟที่ค้นพบเปรียบเทียบกับรูปแบบที่ทำการฝังตัวลงไปว่าตรงตำแหน่งเดียวกันเป็นอัตรากี่เปอร์เซนต์ โดยจะคำนวณส่วนของลำดับย่อยที่ซ้อนทับกันระหว่างโมทีฟที่ค้นพบกับรูปแบบที่ทำการฝังตัวลงไป โดยสามารถอธิบายในรูปของสมการได้ ดังนี้

กำหนดให้ $M^i_{w_i} = (S_{L_{1i}}{}^{w_i}, S_{L_{2i}}{}^{w_i})$ เป็นผลลัพธ์ของโมทีฟทั้งหมดในเซต *i* และ $M^j_{w_j} = (S_{L_{1j}}{}^{w_j}, S_{L_{2j}}{}^{w_j})$ เป็นรูปแบบที่น่าสนใจที่ฝังตัวลงไปทั้งหมดในเซต *j* แล้ว จะสามารถอธิบายค่าของ AoD ได้ดังสมการ

$$AoD(M^i_{w_i}, M^j_{w_j}) = \frac{O(S_{L_{1i}}{}^{w_i}, S_{L_{2i}}{}^{w_i}) + O(S_{L_{1j}}{}^{w_j}, S_{L_{2j}}{}^{w_j})}{U(S_{L_{1i}}{}^{w_i}, S_{L_{2j}}{}^{w_i}) + U(S_{L_{1j}}{}^{w_j}, S_{L_{2j}}{}^{w_j})} \tag{4.1}$$

โดยค่าของ $O(S_{L_x}{}^{w_x}, S_{L_y}{}^{w_y})$ คือ ส่วนที่ซ้อนทับกันของลำดับย่อย 2 ลำดับ และ $U(S_{L_x}{}^{w_x}, S_{L_y}{}^{w_y})$ คือ ส่วนที่ยูเนียนของลำดับย่อย 2 ลำดับ ซึ่งแสดงได้ดังสมการ

$$O(S_{L_x}{}^{w_x}, S_{L_y}{}^{w_y}) = \min(L_x + w_x, L_y + w_y) - \max(L_x, L_y) + 1 \tag{4.2}$$

$$U(S_{L_x}{}^{w_x}, S_{L_y}{}^{w_y}) = \max(L_x + w_x, L_y + w_y) - \min(L_x, L_y) + 1 \quad (4.3)$$

การใช้ AoD ในการวัดผลของโมทีฟที่เป็นผลลัพธ์จากอัลกอริทึม *k*BMD นี้ จะทำโดยการเลือกโมทีฟแรกที่ค้นพบตรงกับรูปแบบที่ฝังตัวลงไปจากเซตของผลลัพธ์โมทีฟที่ดีที่จัดอันดับไว้มาทำการวัดคุณภาพ แล้วเปรียบเทียบกับคุณภาพที่ได้ของ MDS

4.2.2 Accuracy-on-Retrieval (AoR) เป็นวิธีการวัดประสิทธิภาพของฟังก์ชันการให้คะแนนของ *k*BMD โดยความหมายของวิธีการนี้ คือ ถ้ามีรูปแบบที่น่าใจฝังตัวอยู่ จำนวน p รูปแบบ อัลกอริทึมควรจะสามารถจัดอันดับให้โมทีฟที่ค้นพบตรงกับรูปแบบที่ฝังตัวลงไป อยู่ในอันดับที่ 1 ถึง p จึงจะมีประสิทธิภาพที่ดี ดังนั้น ถ้าอัลกอริทึมสามารถจัดอันดับโมทีฟที่ค้นพบตรงกับรูปแบบที่ฝังตัวลงไป ให้อยู่ในอันดับ 1 ถึง p ได้จำนวน opt รูปแบบ จะได้ค่าของ $\text{AoR} = \frac{opt}{p} \times 100$ เปอร์เซ็นต์ โดยวิธีการวัดผลนี้สามารถอธิบายในรูปของสมการได้ ดังนี้

$$\text{AoR} = \frac{opt}{p} \times 100 \quad (4.4)$$

โดย p คือ จำนวนคู่ของรูปแบบที่ทำการฝังตัวลงไปในชุดข้อมูลทดลองและ opt คือ จำนวนโมทีฟที่ดี k^{th}-BM_w ในเซตผลลัพธ์โมทีฟจากอัลกอริทึม *k*BMD โดยที่ $k \leq p$

4.2.3 Reduced Percentage (RP) วิธีการคำนวณหาความสามารถในการลดจำนวนคำตอบของโมทีฟทั้งหมดนั้น เริ่มต้นด้วยการหาจำนวนโมทีฟทั้งหมดตามวิธีการของอัลกอริทึม *k*BMD ซึ่งอยู่ในขั้นตอนการค้นพบโมทีฟทั้งหมด เมื่อได้ค่าของจำนวนโมทีฟทั้งหมดที่สามารถค้นพบได้ในแต่ละข้อมูลอนุกรมเวลา (All Discovered Motif - *ADM*) แล้วจะนำมาเปรียบเทียบกับจำนวนของโมทีฟที่ดี k^{th}-BM_w ในเซตผลลัพธ์โมทีฟจากอัลกอริทึม *k*BMD จำนวน opt โมทีฟ จากนั้นจึงคำนวณค่าเป็นเปอร์เซ็นต์ของจำนวนคำตอบที่สามารถลดได้ออกมา ซึ่งสามารถอธิบายให้อยู่ในรูปของสมการได้ ดังนี้

$$ADM = \sum_{i=2}^{\left\lfloor \frac{|T|}{2} \right\rfloor} DM_i \quad (4.5)$$

โดยค่าของจำนวนโมทีฟที่สามารถค้นพบได้จากอัลกอริทึม *k*BMD ในข้อมูลอนุกรมเวลาข้อมูลหนึ่ง (*ADM*) สามารถหาได้จากผลรวมของค่าจำนวนโมทีฟที่สามารถค้นพบได้ในแต่ละความยาว (Discovered Motif - DM_w)

ค่าของความยาวที่เป็นไปได้ทั้งหมด คือ ความยาวตั้งแต่ 2 ไปจนถึงครึ่งหนึ่งของขนาดข้อมูลอนุกรมเวลา T ดังนั้นจึงต้องรวมค่า DM_i ทั้งหมดโดย i มีค่าตั้งแต่ 2 ไปจนถึง $\left\lfloor \frac{|T|}{2} \right\rfloor$ และค่าของจำนวนโมทีฟที่สามารถค้นพบได้ในแต่ละความยาว DM_w สามารถหาได้จากขั้นตอนการค้นพบโมทีฟของอัลกอริทึม kBMD ซึ่งหมายถึงจำนวนของ k^{th}-Time Series Motif ที่สามารถค้นพบได้ในแต่ละความยาวโมทีฟนั้น ๆ

จากนั้น เมื่อได้ค่าของจำนวนโมทีฟที่สามารถค้นพบได้ในข้อมูลอนุกรมเวลา T คือ ADM แล้ว จะคำนวณหาความสามารถในการลดจำนวนคำตอบโมทีฟ ได้ดังสมการ

$$\mathrm{RP} = \frac{ADM - opt}{ADM} \times 100 \tag{4.6}$$

เมื่อ opt คือ จำนวนโมทีฟที่ดี k^{th}-BM_w ในเซตผลลัพธ์โมทีฟของอัลกอริทึม kBMD

ในลำดับถัดไปจะเป็นผลการทดลองทั้งหมด โดยใช้วิธีการวัดผลและประเมินผลที่ได้นำเสนอไปในหัวข้อนี้

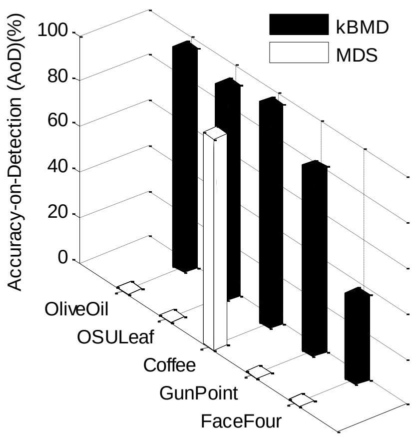
4.3 ผลการทดลองและวิเคราะห์ผลการทดลอง

ในหัวข้อนี้ จะนำเสนอผลการทดลองตามลำดับตั้งแต่ขั้นตอนการวัดคุณภาพของโมทีฟโดยใช้ Accuracy-on-Detection (AoD) จากนั้นจึงเป็นประสิทธิภาพของฟังก์ชันการให้คะแนนในการจัดอันดับโมทีฟที่ดีโดยใช้ Accuracy-on-Retrieval (AoR) และสุดท้าย คือ ความสามารถในการลดจำนวนคำตอบโมทีฟที่เป็นไปได้ Reduced Percentage (RP) รวมไปถึงวิเคราะห์ผลการทดลองทั้งหมดของแต่ละขั้นตอน โดยผลการทดลองทั้งหมดมีดังต่อไปนี้

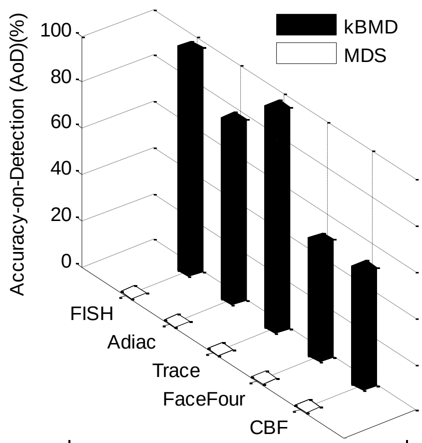
4.3.1 คุณภาพโมทีฟของอัลกอริทึม *k*BMD เปรียบเทียบกับ MDS โดยใช้ Accuracy-on-Detection (AoD)(%)

เนื่องจากวิธีการของงานวิจัยที่เกี่ยวข้อง (Motif Discovery based on SAX - MDS) [11] นั้น มีพารามิเตอร์ความยาวโมทีฟเริ่มต้น (w) ดังนั้น ผู้วิจัยจึงทำการทดลองเปรียบเทียบด้วยการให้ค่าพารามิเตอร์ความยาวที่เป็นค่าที่ดีที่สุดสำหรับอัลกอริทึม MDS โดยกำหนดให้ค่า w เท่ากับความยาวของรูปแบบที่ทำการฝังตัวลงไปแต่ละรูปแบบ โดยทำการประมวลผลซ้ำเท่ากับจำนวนของรูปแบบที่ฝังตัวในชุดข้อมูลทดลอง เพื่อเปลี่ยนค่าของความยาวให้

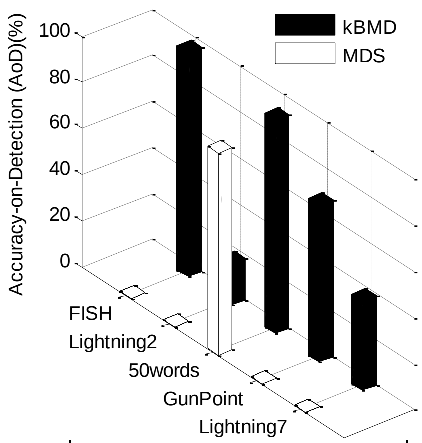
เท่ากับรูปแบบที่ฝังตัวแต่ละรูปแบบ ซึ่งผลการทดลองของ *k*BMD เปรียบเทียบกับ MDS ในชุดข้อมูลทดลองส่วนที่ 1 จำนวน 6 ชุดข้อมูลเป็นดังต่อไปนี้



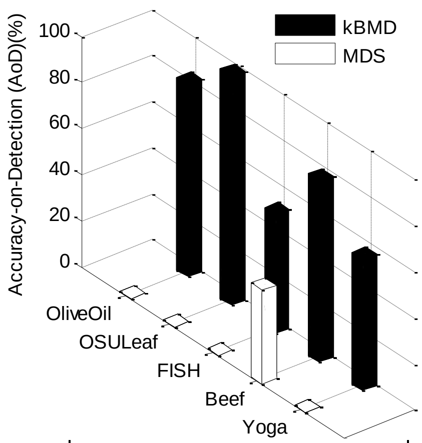
ภาพที่ 4.3 ผลการทดลองชุดข้อมูลที่ 1



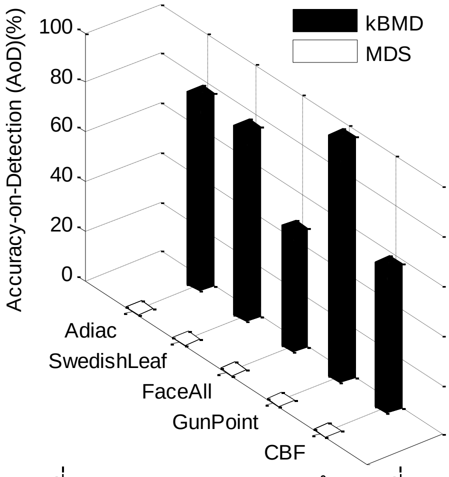
ภาพที่ 4.4 ผลการทดลองชุดข้อมูลที่ 2



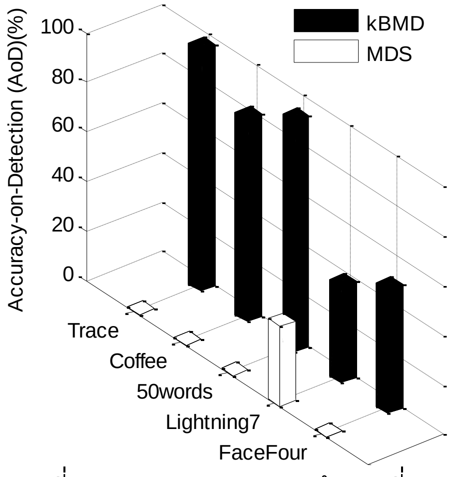
ภาพที่ 4.5 ผลการทดลองชุดข้อมูลที่ 3



ภาพที่ 4.6 ผลการทดลองชุดข้อมูลที่ 4

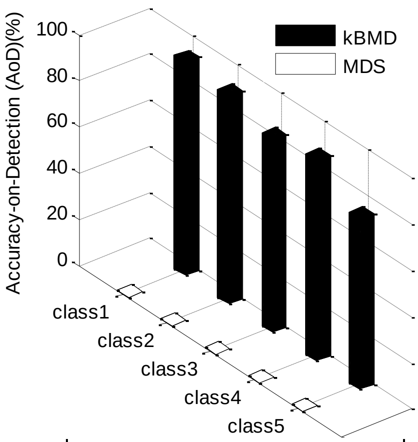


ภาพที่ 4.7 ผลการทดลองชุดข้อมูลที่ 5

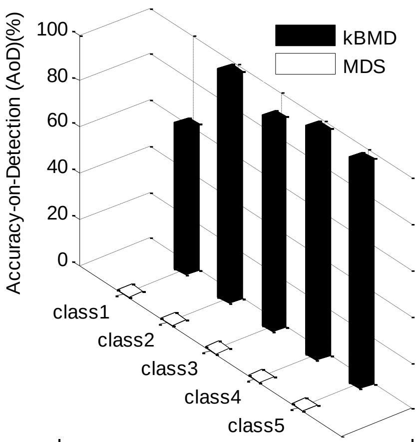


ภาพที่ 4.8 ผลการทดลองชุดข้อมูลที่ 6

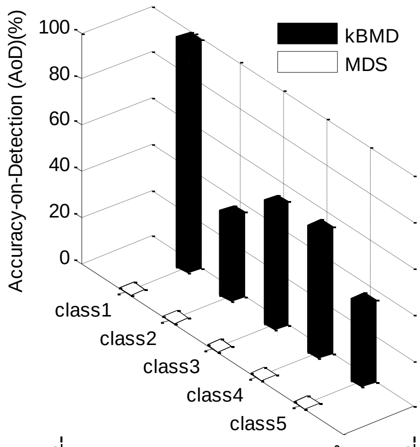
ส่วนของผลการทดลองของ *k*BMD เปรียบเทียบกับ MDS ในชุดข้อมูลทดลองส่วนที่ 2 จำนวน 10 ชุดข้อมูลเป็นดังต่อไปนี้



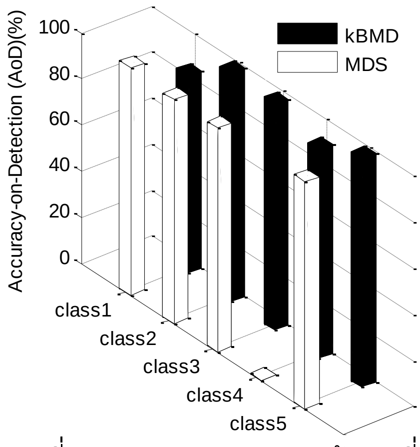
ภาพที่ 4.9 ผลการทดลองชุดข้อมูลที่ 1



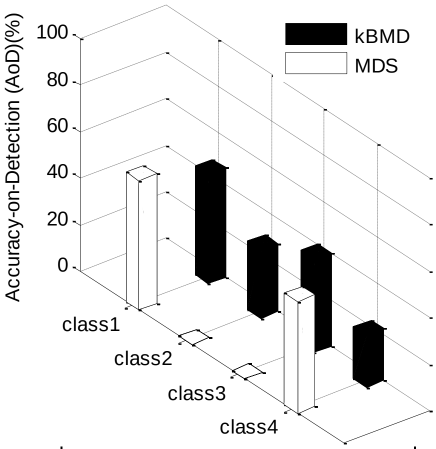
ภาพที่ 4.10 ผลการทดลองชุดข้อมูลที่ 2



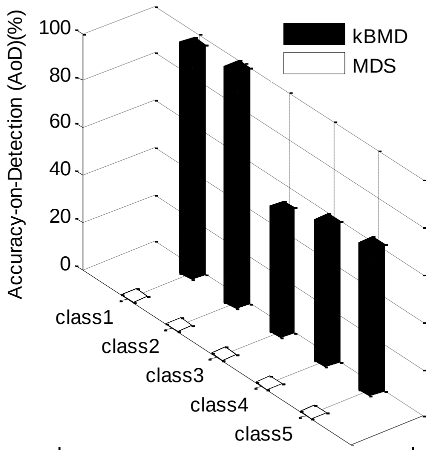
ภาพที่ 4.11 ผลการทดลองชุดข้อมูลที่ 3



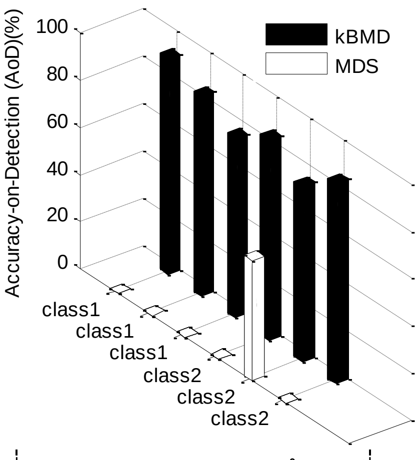
ภาพที่ 4.12 ผลการทดลองชุดข้อมูลที่ 4



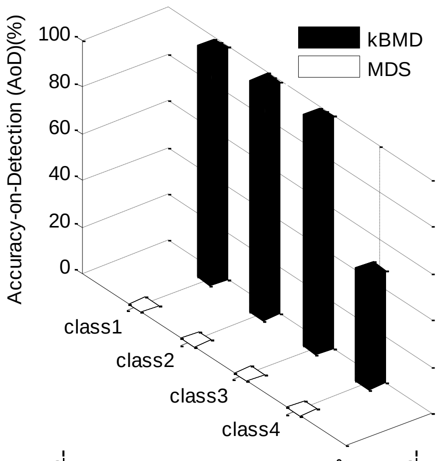
ภาพที่ 4.13 ผลการทดลองชุดข้อมูลที่ 5



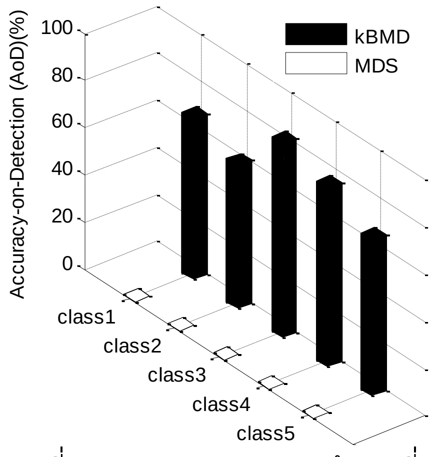
ภาพที่ 4.14 ผลการทดลองชุดข้อมูลที่ 6



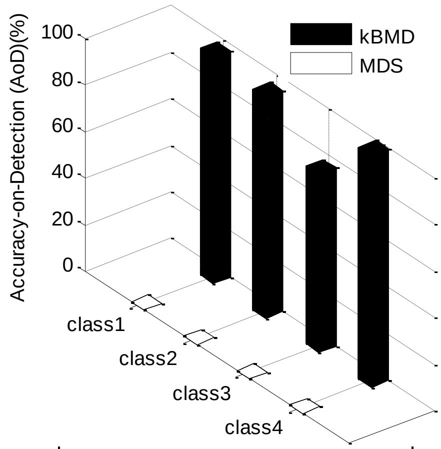
ภาพที่ 4.15 ผลการทดลองชุดข้อมูลที่ 7 ฝังตัวรูปแบบ GunPoint ที่มี 2 ประเภท คือ Gun 6 ลำดับและ Point 6 ลำดับ (อย่างละ 3 คู่)



ภาพที่ 4.16 ผลการทดลองชุดข้อมูลที่ 8



ภาพที่ 4.17 ผลการทดลองชุดข้อมูลที่ 9



ภาพที่ 4.18 ผลการทดลองชุดข้อมูลที่ 10

จากผลการทดลองทั้งหมดนี้ อัลกอริทึม *k*BMD สามารถค้นพบรูปแบบที่ทำการฝังตัวลงไปได้ครบทุกรูปแบบโดยมี AoD สูงกว่าอัลกอริทึม MDS ที่นำมาเปรียบเทียบ แต่อัลกอริทึม MDS จะสามารถค้นพบรูปแบบได้เป็นส่วนน้อยเท่านั้น ซึ่งจากผลการทดลองทั้งหมดดังภาพที่แสดงไว้ข้างต้น กราฟที่มีค่า AoD เป็น 0 ของ MDS หมายถึง MDS ไม่สามารถค้นพบรูปแบบนั้น ๆ ได้ ทั้งนี้เนื่องมาจากอัลกอริทึม MDS ใช้วิธีการลดจำนวนมิติของข้อมูลโดยแปลงแต่ละลำดับย่อยข้อมูลอนุกรมเวลาให้เป็นตัวอักษร ด้วยมิติข้อมูลที่ลดลงอย่างมากทำให้คุณลักษณะบางอย่างของลำดับย่อยอาจขาดหายไป แม้ให้ค่าพารามิเตอร์ความยาวโมทีฟที่ดีที่สุดแล้ว การเปรียบเทียบความคล้ายกันของลำดับย่อยยังให้ค่าที่ผิดพลาดมากเนื่องจากเมื่อทำการลดมิติข้อมูลแล้ว ข้อมูลแบบสุ่มจะมีโอกาสที่จะถูกแปลงเป็นตัวอักษรได้เหมือนกับลำดับย่อยที่เป็นข้อมูลแบบสุ่มที่ตำแหน่งอื่น ๆ ในข้อมูลอนุกรมเวลาหรือคล้ายกับรูปแบบที่ฝังตัวลงไปทำให้การค้นพบโมทีฟนั้นผิดพลาด โดยมีชุดข้อมูลทดลองเป็นจำนวนมากที่อัลกอริทึม MDS ไม่สามารถค้นพบรูปแบบที่ทำการฝังตัวลงไปนั้นออกมาได้เลยเนื่องจาก ข้อมูลแบบสุ่มถูกแปลงเป็นตัวอักษรแล้วให้รูปแบบที่เหมือนกันในจำนวนที่มากกว่า สิ่งที่ค้นพบจึงเป็นข้อมูลแบบสุ่มเป็นส่วนใหญ่

ชุดข้อมูลทดลองที่ให้ผลที่น่าสนใจส่วนหนึ่งคือ ชุดข้อมูลที่มีรูปแบบ Coffee ฝังตัวลงไปดังภาพที่ 4.3 และภาพที่ 4.12 สังเกตว่าเป็นรูปแบบหนึ่งที่อัลกอริทึม MDS สามารถค้นพบออกมาเป็นโมทีฟได้ในค่า AoD ที่สูง ในส่วนนี้มีปัจจัยซึ่งขึ้นอยู่กับข้อมูลแบบสุ่มเป็นส่วนหนึ่งที่เมื่อทำการแปลงเป็นตัวอักษรแล้วมีรูปแบบที่เหมือนกันจำนวนน้อยกว่า และรูปแบบ Coffee ที่แปลงเป็นตัวอักษรแล้วมีความเหมือนกันเป็นจำนวนมากกว่า รูปแบบ Coffee จึงถูกค้นพบออกมาและมีค่า AoD ที่สูงจากการให้ค่าของพารามิเตอร์ความยาวที่ดีที่สุดคือให้ค่าเท่ากับขนาดของรูปแบบที่ทำการฝังตัวลงไป อีกชุดข้อมูลหนึ่ง คือ ผลการทดลองดังภาพที่ 4.13 จะสังเกตว่า MDS นั้นให้ค่า

ของ AoD ที่สูงกว่าเช่นกัน ทั้งนี้เนื่องมาจากข้อมูลที่ฝังตัวลงไปในชุดข้อมูลนี้ คือรูปแบบ FaceFour ซึ่งมีส่วนของหน้า ที่มีความคล้ายกันสูงแต่ส่วนที่เป็นผมจะมีข้อมูลรบกวนมาก *k*BMD จึงสามารถค้นพบเพียงส่วนหน้าของ FaceFour แล้วคัดโมทีฟที่รวมส่วนผมออกไป เนื่องจากมีค่าของระยะทางยุคลิดสูง ส่วนอัลกอริทึม MDS ที่สามารถค้นพบโมทีฟได้ในค่า AoD ที่สูงกว่าเนื่องจาก MDS ทำการลดมิติข้อมูลก่อนการเปรียบเทียบทำให้ MDS ทนต่อข้อมูลรบกวนได้ดีกว่า แต่อย่างไรก็ตาม โมทีฟที่ค้นพบโดย MDS นั้นมีเพียง 2 ประเภทจาก 4 ประเภทและค่า AoD ที่สูงกว่าส่วนหนึ่งยังเนื่องมาจากการกำหนดค่าความยาวเริ่มต้นที่ดีที่สุดด้วย สุดท้ายคือผลการทดลองดังภาพที่ 4.15 ที่แสดงเป็น 3 คู่ลำดับต่อ 1 ประเภท เนื่องจากเป็นชุดข้อมูลที่ฝังตัวข้อมูล GunPoint จึงมีเพียง 2 ประเภท คือ ประเภท Gun และ ประเภท Point โดยทำการฝังตัวลงไปประเภทละ 6 ลำดับ เป็นจำนวน 3 คู่ โดย MDS สามารถค้นพบได้เพียง 1 คู่ ซึ่งเปรียบเทียบกันโดยตรงกับคู่ที่ *k*BMD ค้นพบซึ่งเป็นคู่เดียวกัน การแสดงผลการทดลองจึงเป็น 3 คู่ลำดับต่อ 1 ประเภท

ลำดับถัดไปจะนำเสนอผลการวัดประสิทธิภาพของฟังก์ชันการให้คะแนนของอัลกอริทึม *k*BMD

4.3.2 ประสิทธิภาพของฟังก์ชันการให้คะแนนของอัลกอริทึม *k*BMD โดยใช้ Accuracy-on-Retrieval (AoR)(%)

ผลการทดลองของการวัดประสิทธิภาพของฟังก์ชันการให้คะแนนของอัลกอริทึม *k*BMD กับชุดข้อมูลทดลองส่วนที่ 1 เป็นดังตารางที่ 4.3

ตารางที่ 4.3 ผลการทดลองของการวัดประสิทธิภาพของฟังก์ชันการให้คะแนนของอัลกอริทึม *k*BMD กับชุดข้อมูลทดลองส่วนที่ 1

ชุดข้อมูล	AoR (%)	**ชุดข้อมูล**	AoR (%)	**ชุดข้อมูล**	AoR (%)
ชุดข้อมูล 1	100	ชุดข้อมูล 2	80	ชุดข้อมูล 3	80
ชุดข้อมูล 4	100	ชุดข้อมูล 5	60	ชุดข้อมูล 6	80

ผลการทดลองของการวัดประสิทธิภาพของฟังก์ชันการให้คะแนนของอัลกอริทึม *k*BMD กับชุดข้อมูลทดลองส่วนที่ 2 เป็นดังตารางที่ 4.4

ตารางที่ 4.4 ผลการทดลองของการวัดประสิทธิภาพของฟังก์ชันการให้คะแนนของอัลกอริทึม *k*BMD กับชุดข้อมูลทดลองส่วนที่ 2

ชุดข้อมูล	AoR (%)	**ชุดข้อมูล**	AoR (%)	**ชุดข้อมูล**	AoR (%)
ชุดข้อมูล 1	100	ชุดข้อมูล 2	100	ชุดข้อมูล 3	80
ชุดข้อมูล 4	80	ชุดข้อมูล 5	75	ชุดข้อมูล 6	100
ชุดข้อมูล 7	100	ชุดข้อมูล 8	75	ชุดข้อมูล 9	100
ชุดข้อมูล 10	75				

จากผลการวัดประสิทธิภาพของฟังก์ชันการให้คะแนนของอัลกอริทึม *k*BMD ของชุดข้อมูลทั้งหมด ผลลัพธ์ที่ได้อยู่ในค่าของ AoR ที่สูง โดยโมทีฟที่ดีที่ค้นพบในอันดับที่ $k \leq N$ แต่ไม่ตรงกับรูปแบบที่ฝังตัวลงไปนั้นยังคงเป็นโมทีฟที่มีความคล้ายกันทางรูปร่าง ซึ่งส่วนใหญ่จะเป็นส่วนหนึ่งของโมทีฟที่ทำการฝังตัวลงไปแต่เนื่องจากเป็นคู่ของลำดับย่อยที่ไม่ตรงกับคู่ของรูปแบบที่ทำการฝังตัวลงไปจึงไม่สามารถนำมาคิดเป็นโมทีฟที่อยู่ในเกณฑ์อันดับที่ $k \leq N$ ได้ และอีกกรณีหนึ่งที่ทำให้ค่า AoR ลดลงคือ การฝังตัวรูปแบบเดียวที่ความยาวเดียวในชุดข้อมูลส่วนที่ 2 ซึ่งรูปแบบมีความคล้ายกันทางรูปร่างแต่อยู่คนละประเภท (Class) ซึ่งเป็นส่วนหนึ่งที่ทำให้อัลกอริทึม *k*BMD ค้นพบออกมาเป็นโมทีฟแต่คู่ของลำดับย่อยของโมทีฟนั้นอยู่คนละประเภทจึงไม่สามารถนำมานับรวมในเกณฑ์อันดับที่ $k \leq N$ ได้เช่นกัน แต่อย่างไรก็ตามฟังก์ชันการให้คะแนนนี้ยังสามารถจัดอันดับของโมทีฟที่ตรงตามการวัดผลออกมาได้ในค่า AoR ที่สูง

ลำดับถัดไปจะนำเสนอผลการวัดความสามารถในการลดจำนวนคำตอบโมทีฟที่เป็นไปได้ทั้งหมดของอัลกอริทึม *k*BMD เพื่อช่วยสนับสนุนการประสิทธิภาพของอัลกอริทึม

4.3.3 **ความสามารถในการลดจำนวนคำตอบโมทีฟที่เป็นไปได้ทั้งหมดของอัลกอริทึม *k*BMD โดยใช้** Reduced Percentage (RP)(%)

ผลการทดลองของการวัดความสามารถในการลดจำนวนคำตอบโมทีฟที่เป็นไปได้ทั้งหมดของอัลกอริทึม *k*BMD กับชุดข้อมูลทดลองส่วนที่ 1 เป็นดังตารางที่ 4.5

ตารางที่ 4.5 ผลการทดลองของการวัดความสามารถในการลดจำนวนคำตอบโมทีฟที่เป็นไปได้ทั้งหมดของอัลกอริทึม *k*BMD กับชุดข้อมูลทดลองส่วนที่ 1

ชุดข้อมูล	RP (%)	**ชุดข้อมูล**	RP (%)	**ชุดข้อมูล**	RP (%)
ชุดข้อมูล 1	99.88	ชุดข้อมูล 2	99.81	ชุดข้อมูล 3	99.72
ชุดข้อมูล 4	99.98	ชุดข้อมูล 5	99.62	ชุดข้อมูล 6	99.67

ผลการทดลองของการวัดความสามารถในการลดจำนวนคำตอบโมทีฟที่เป็นไปได้ทั้งหมดของอัลกอริทึม *k*BMD กับชุดข้อมูลทดลองส่วนที่ 2 เป็นดังตารางที่ 4.6

ตารางที่ 4.6 ผลการทดลองของการวัดความสามารถในการลดจำนวนคำตอบโมทีฟที่ค้นพบได้ทั้งหมดของอัลกอริทึม *k*BMD กับชุดข้อมูลทดลองส่วนที่ 2

ชุดข้อมูล	RP (%)	**ชุดข้อมูล**	RP (%)	**ชุดข้อมูล**	RP (%)
ชุดข้อมูล 1	99.64	ชุดข้อมูล 2	99.63	ชุดข้อมูล 3	99.79
ชุดข้อมูล 4	99.72	ชุดข้อมูล 5	99.61	ชุดข้อมูล 6	99.89
ชุดข้อมูล 7	99.62	ชุดข้อมูล 8	99.92	ชุดข้อมูล 9	99.52
ชุดข้อมูล 10	99.74				

จากผลการทดลองการวัดความสามารถในการลดจำนวนคำตอบโมทีฟที่ค้นพบทั้งหมด แสดงให้เห็นว่าอัลกอริทึม *k*BMD สามารถลดจำนวนคำตอบโมทีฟที่ค้นพบทั้งหมดลงเหลือไม่ถึง 1 เปอร์เซ็นต์ออกมาเป็นเซตคำตอบ “โมทีฟที่ดี” ซึ่งสามารถสนับสนุนประสิทธิภาพของอัลกอริทึม *k*BMD ได้เป็นอย่างดีว่าผลการทดลองที่ได้จากการวัดคุณภาพโมทีฟและการจัดอันดับความดีของโมทีฟยังสามารถค้นพบรูปแบบที่ฝังตัวลงไปได้ในค่า AoD และ AoR ที่สูง แม้ว่าจะมีจำนวนโมทีฟจำนวนมาก ในลำดับถัดไปจะเป็นการสรุปผลการทดลองทั้งหมดที่ได้นำเสนอในบทนี้

4.4 สรุปผลการทดลอง

จากการวัดผลการทดลองทั้ง 3 ขั้นตอนที่ได้นำเสนอมาทั้งหมดจะสรุปได้ว่าอัลกอริทึม *k*BMD สามารถค้นพบโมทีฟโดยไม่ต้องกำหนดพารามิเตอร์ความยาว ได้ด้วยคุณภาพ AoD และ AoR ที่สูง และเมื่อเปรียบเทียบกับอัลกอริทึม MDS โมทีฟที่ได้จากอัลกอริทึม *k*BMD จะ

มีคุณภาพ AoD ที่สูงกว่า และยังสามารถลดจำนวนคำตอบโมทีฟที่เป็นไปได้ทั้งหมดในร้อยละที่สูงเกิน 99 เปอร์เซ็นต์

สรุปในบทที่ 4 นี้ ได้นำเสนอการทดลองเพื่อประเมินประสิทธิภาพอัลกอริทึม *k*BMD ที่นำเสนอในวิทยานิพนธ์นี้ เริ่มจากการเตรียมข้อมูลทดลองและการวัดผลซึ่งประกอบไปด้วย 3 ขั้นตอน คือ การวัดคุณภาพของโมทีฟที่ค้นพบโดยใช้ AoD การวัดประสิทธิภาพของฟังก์ชันการให้คะแนนของอัลกอริทึม *k*BMD โดยใช้ AoR และการวัดความสามารถในการลดจำนวนคำตอบของโมทีฟที่เป็นไปได้ทั้งหมดของอัลกอริทึม *k*BMD รวมทั้งการวิเคราะห์และสรุปผล ในลำดับถัดไปจะเป็นการสรุปผลการวิจัย อภิปรายผลการทดลองและข้อเสนอแนะ

บทที่ 5
สรุปผลการวิจัย อภิปรายผลและข้อเสนอแนะ

งานวิจัยนี้ นำเสนออัลกอริทึมในการค้นพบโมทีฟที่ดีที่เรียกว่า *k*-Best Motif Discovery (*k*BMD) ซึ่งเป็นอัลกอริทึมแรกที่ไม่ต้องกำหนดพารามิเตอร์ใด ๆ เพิ่มเติมโดยเฉพาะอย่างยิ่งความยาวโมทีฟที่เป็นพารามิเตอร์ที่กำหนดค่าได้ยากเนื่องจากจำนวนโมทีฟทั้งหมดที่เป็นไปได้นั้นมีจำนวนมาก อัลกอริทึมแก้ปัญหาการกำหนดค่าความยาวโมทีฟโดยการค้นพบโมทีฟทั้งหมดแล้วทำการลดจำนวนโมทีฟทั้งหมดลงตามลำดับขั้นตอนไปจนถึงการให้คะแนนโมทีฟเพื่อจัดอันดับของ “โมทีฟที่ดี” ออกมาเป็นเซตคำตอบของอัลกอริทึม ซึ่งเป็นการเปลี่ยนปัญหาของการกำหนดความยาวโมทีฟมาเป็นปัญหาของการเลือก k^{th}-Best Motif ที่ง่ายกว่ามาก โดยอัลกอริทึมสามารถค้นพบรูปแบบที่ทำการฝังตัวลงไปในข้อมูลอนุกรมเวลาออกมาเป็นผลลัพธ์โมทีฟได้อย่างถูกต้องครอบคลุมรูปแบบที่ทำการฝังตัวลงไปทั้งหมดด้วยคุณภาพของโมทีฟที่สูงและเมื่อเปรียบเทียบกับอัลกอริทึมของงานวิจัยที่เกี่ยวข้อง (MDS) ซึ่งยังต้องกำหนดพารามิเตอร์ความยาวเริ่มต้น แม้ในการทดลองจะกำหนดค่าที่ดีที่สุดให้อัลกอริทึม MDS แล้ว อัลกอริทึม kBMD ที่นำเสนอในงานวิจัยชิ้นนี้ก็ยังสามารถให้คุณภาพของผลลัพธ์โมทีฟที่ดีกว่ามาก ดังที่ได้นำเสนอผลการทดลองและวิเคราะห์ผลการทดลองไว้ในบทที่ผ่านมา โดยสามารถสรุปผลการวิจัยทั้งหมดได้ดังนี้

5.1 สรุปและอภิปรายผลการวิจัย

ในหัวข้อนี้จะสรุปและอภิปรายงานวิจัยทั้งหมดตั้งแต่จุดเริ่มต้นของการวิจัยตามด้วยอัลกอริทึมที่นำเสนอไปจนถึงสรุปผลการทดลอง ดังนี้

งานวิจัยชิ้นนี้ เน้นที่การค้นพบโมทีฟของข้อมูลอนุกรมเวลา โดยเมื่อเริ่มทำการค้นหาโมทีฟจะต้องกำหนดพารามิเตอร์ความยาวโมทีฟทุกครั้งเพื่อสกัดแต่ละลำดับย่อยออกมาทำการเปรียบเทียบ โดยพารามิเตอร์ความยาวนี้ถ้ามีการเปลี่ยนแปลง โมทีฟที่เป็นผลลัพธ์จะแตกต่างกันซึ่งเป็นการยากที่จะเลือกมาใช้งาน ปัญหานี้จึงเป็นแรงจูงใจของงานวิจัยซึ่งถ้าสามารถทำการค้นพบโมทีฟได้โดยไม่ต้องกำหนดความยาวโมทีฟได้ จะทำให้การค้นหาโมทีฟง่ายขึ้นอย่างมาก

งานวิจัยที่ผ่านมาที่ทำการแก้ปัญหาความยาวโมทีฟ ยังไม่สามารถแก้ปัญหานี้ได้อย่างแท้จริงเนื่องจากยังต้องกำหนดค่าพารามิเตอร์ความยาวโมทีฟเช่นเดิม โดยเป็นความยาวเริ่มต้น รวมไปถึงพารามิเตอร์อื่นที่เพิ่มเติมขึ้นมา ซึ่งจากการทดลองแล้วต้องใกล้เคียงกับรูปแบบที่

น่าสนใจในข้อมูลอนุกรมเวลาจึงจะมีโอกาสที่จะค้นพบโมทีฟที่ตรงกับรูปแบบที่น่าสนใจนั้นออกมา ปัญหาความยาวโมทีฟจึงยังไม่ได้รับการแก้ไข ดังนั้น งานวิจัยที่นำเสนอนี้จึงเป็นงานวิจัยชิ้นแรกที่ทำการแก้ปัญหาความยาวโมทีฟ

วิธีการของอัลกอริทึมที่นำเสนอเรียกว่า อัลกอริทึมการค้นพบโมทีฟที่ดี *k*-Best Motif Discovery (*k*BMD) ที่มีข้อมูลอนุกรมเวลาเป็นข้อมูลนำเข้าโดยไม่มีพารามิเตอร์ใด ๆ เพิ่มเติม โดยมีขั้นตอนหลักอยู่ 4 ขั้นตอนของการค้นหาโมทีฟทั้งหมด คือ การค้นพบโมทีฟ การแบ่งกลุ่มโมทีฟ การเลือกตัวแทนกลุ่มโมทีฟและการคำนวณคะแนนเพื่อจัดอันดับ “โมทีฟที่ดี” โดยอัลกอริทึมให้ผลลัพธ์เป็นเซตของ “โมทีฟที่ดี” ที่จัดอันดับไว้ให้ผู้ใช้เลือกไปใช้งาน

การทดลองเพื่อวัดประสิทธิภาพของอัลกอริทึม *k*BMD แบ่งออกเป็น 3 ขั้นตอน คือ การวัดคุณภาพของผลลัพธ์โมทีฟเพื่อเปรียบเทียบกับอัลกอริทึมของงานวิจัยที่เกี่ยวข้อง (MDS) โดยใช้ Accuracy-on-Detection (AoD) การวัดประสิทธิภาพของฟังก์ชันการให้คะแนนของอัลกอริทึม *k*BMD โดยใช้ Accuracy-on-Retrieval (AoR) และการวัดความสามารถในการลดจำนวนคำตอบโมทีฟที่เป็นไปได้ Reduced Percentage (RP) โดยผลการทดลองสรุปได้ว่าอัลกอริทึม *k*BMD สามารถค้นพบโมทีฟได้ในคุณภาพของโมทีฟที่ค่าของ AoD ที่สูงและสูงกว่าผลลัพธ์โมทีฟจากอัลกอริทึม MDS อย่างมาก ในส่วนของประสิทธิภาพของฟังก์ชันการให้คะแนนของอัลกอริทึม *k*BMD จะให้ค่า AoR ที่สูงเช่นกัน โดยเมื่อดูจากผลของการวัดความสามารถในการลดจำนวนโมทีฟที่เป็นไปได้ซึ่งให้ค่าของ RP สูงกว่า 99 เปอร์เซ็นต์แล้วเป็นข้อสนับสนุนได้อย่างดีว่าแม้จำนวนคำตอบโมทีฟจะมีจำนวนมากแต่การจัดอันดับของอัลกอริทึมก็ยังให้โมทีฟที่ตรงกับรูปแบบที่ฝังตัวลงไปด้วยคุณภาพ AoD และ AoR ที่สูง

5.2 ข้อจำกัดและข้อเสนอแนะ

แม้ว่าจากการทดลอง *k*BMD จะให้คุณภาพของโมทีฟ AoD ที่สูงกว่าวิธีการของงานวิจัยที่ผ่านมาที่นำมาเปรียบเทียบและสามารถค้นพบรูปแบบที่ฝังตัวลงไปได้ทั้งหมดโดยให้ผลลัพธ์ของการจัดอันดับโมทีฟที่มีสามารถค้นพบรูปแบบที่ฝังตัวอยู่ในอันดับต้น ๆ ได้ค่า AoR ที่สูง รวมไปถึงมีความสามารถในการลดจำนวนคำตอบของโมทีฟที่สูงมาก แต่ในวิธีการของงานวิจัยนี้ยังมีข้อจำกัดบางประการเนื่องจากในขั้นตอนของ *k*BMD จะต้องทำการค้นหาโมทีฟทั้งหมดที่ทุกความยาวที่เป็นไปได้ นำมาซึ่งปัญหาของเวลาในการประมวลผล โดย Big-O ของขั้นตอนการค้นพบโมทีฟนี้จะมีค่าเท่ากับ $O(mn^2)$ เมื่อ m คือจำนวนของค่าความยาวโมทีฟที่เป็นไปได้และ n คือ จำนวนของลำดับย่อยทั้งหมดที่เป็นไปได้ในข้อมูลอนุกรมเวลา T ในแต่ละความยาวโมทีฟ ดังนั้น เวลาในการประมวลผลจะเพิ่มขึ้นแปรผันตามขนาดความยาวของข้อมูลอนุกรมเวลา T

เนื่องจากงานวิจัยนี้เป็นงานแรกที่แก้ปัญหาความยาวของโมทีฟจึงหลีกเลี่ยงไม่ได้ที่จะต้องค้นพบโมทีฟทั้งหมดออกมาเพื่อวิเคราะห์ ซึ่งเป็นข้อจำกัดข้อหนึ่งของอัลกอริทึม แต่อย่างไรก็ตามผลลัพธ์โมทีฟของอัลกอริทึมก็ให้คุณภาพที่สูงกว่าวิธีการของงานวิจัยที่ผ่านมามากเพราะงานวิจัยที่ผ่านมาไม่สามารถค้นพบรูปแบบที่ทำการฝังตัวลงไปได้เลยในแต่ละชุดข้อมูลเป็นส่วนใหญ่ ในส่วนของงานวิจัยชิ้นนี้ยังเป็นแนวทางในการพัฒนาต่อเพื่อแก้ไขปัญหาของเวลาการประมวลผลซึ่งมีได้หลายแนวทาง ที่ผ่านมามีงานวิจัยที่พยายามเพิ่มความเร็วของอัลกอริทึมการหาโมทีฟ Mueen-Keogh (MK) ขึ้นไปอีก โดยใช้แนวทางการประมวลผลแบบขนานเพื่อกระจายการประมวลผลของแต่ละส่วนของอัลกอริทึม [15] นอกจากการเพิ่มความเร็วในการประมวลผลแล้วยังรวมไปถึงการเพิ่มคุณภาพของโมทีฟให้ดียิ่ง ๆ ขึ้นไป

โดยทั่วไปแล้วการค้นพบโมทีฟในข้อมูลอนุกรมเวลาเป็นขั้นตอนย่อยของกระบวนการประมวลผลด้านการทำเหมืองข้อมูลหลัก ๆ เช่น การจัดกลุ่มข้อมูล (Clustering) การจำแนกประเภทข้อมูล (Classification) และการหากฎความสัมพันธ์ของข้อมูล (Association Rule) ตัวอย่างของการนำการค้นพบโมทีฟไปใช้งานเช่น ใช้เป็นขั้นตอนเตรียมการก่อนทำการจำแนกประเภทข้อมูลซึ่งจะใช้การค้นพบโมทีฟเพื่อหารูปแบบที่น่าสนใจออกมาจากข้อมูลอนุกรมเวลานั้นก่อน เพื่อนำรูปแบบที่ค้นพบนั้น ๆ ไปเข้าสู่กระบวนการจำแนกประเภทของข้อมูลอีกทีหนึ่ง ดังตัวอย่างที่นำไปใช้ในงานวิจัย [16] การนำโมทีฟไปใช้งานอีกรูปแบบหนึ่งเป็นการนำไปทำการจัดกลุ่มข้อมูลโดยใช้การค้นพบโมทีฟเป็นขั้นตอนแรกที่แบ่งรูปแบบของข้อมูลที่ค้นพบออกเป็นแต่ละรูปแบบเพื่อนำรูปแบบเหล่านี้ไปจัดกลุ่มของข้อมูล ดังตัวอย่างที่นำไปใช้ในงานวิจัย [17] เป็นต้น

รายการอ้างอิง

[1] E. Keogh, X. Xi, L. Wei, and C. A. Ratanamahatana. The UCR Time Series Classification/Clustering Homepage. [Online]. 2008. Available from: www.cs.ucr.edu/~eamonn/time_series_data/ [2008]

[2] P. Beaudoin, S. Coros, M. V. Panne, and P. Poulin. Motion-Motif Graphs. Proceedings of the 2008 ACM SIGGRAPH/Eurographics Symposium on Computer Animation, pp.117-126. 7-9 July 2008. Dublin, Ireland. 2008.

[3] B. Chiu, E. Keogh, and S. Lonardi. Probabilistic Discovery of Time Series Motifs. Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp.493-498. 24 - 27 August 2003. Washington DC, USA. 2003.

[4] T. Guyet, C. Garbay, and M. Dojat. Knowledge Construction from Time Series Data Using a Collaborative Exploration System. Journal of Biomedical Informatics 40 (2007): 672–687.

[5] J. Meng, J. Yuan, M. Hans, and Y. Wu. Mining Motifs from Human Motion. Proceedings of the 29th Annual Conference of the European Association for Computer Graphics, pp.71-74. 14-18 April 2008. Crete, Greece. 2008.

[6] D. Minnen, C. L. Isbell, I. Essa, and T. Starner. Discovering Multivariate Motifs Using Subsequence Density Estimation and Greedy Mixture Learning. Proceedings of the 22nd Conference on Artificial Intelligence, pp.615-620. 22–26 July 2007. Vancouver, British Columbia, Canada. 2007.

[7] Y. Tanaka, K. Iwamoto, and K. Uehara. Discovery of Time-Series Motif from Multi-Dimensional Data Based on MDL Principle. Machine Learning 58 (2005): 269-300.

[8] S. Rombo and G. Terracina. Discovering Representative Models in Large Time Series Databases. Proceedings of the 6th International Conference on

Flexible Query Answering Systems, pp.84-97. 24-26 June 2004. Lyon, France. 2004.

[9] A. Mueen, E. Keogh, Q. Zhu, S. Cash, and B. Westover. Exact Discovery of Time Series Motifs. Proceedings of the 9th SIAM International Conference on Data Mining, pp.473-484. 30 April - 2 May 2009. Sparks, Nevada. 2009.

[10] H. Tang and S. S. Liao. Discovering original motifs with different lengths from time series. Knowledge-Based Systems 21 (2008): 666–671.

[11] Y. Li and J. Lin. Approximate Variable-Length Time Series Motif Discovery Using Grammar Inference. Proceedings of the 10th International Workshop on Multimedia Data Mining, pp.1-9. 25-28 July 2010. Washington DC, U.S.A. 2010.

[12] J. Lin, E. Keogh, S. Lonardi, and P. Patel. Finding Motifs in Time Series. The 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp.53-68. 23-26 July 2002. Edmonton Alberta, Canada. 2002.

[13] J. Lin, E. Keogh, L. Wei, and S. Lonardi. Experiencing SAX: a Novel Symbolic Representation of Time Series. Data Mining and Knowledge Discovery 15 (2007): 107-144.

[14] V. Niennattrakul, D. Wanichsan, and C. A. Ratanamahatana. Accurate Subsequence Matching on Data Stream under Time Warping Distance. The 13th Pacific-Asia Conference on Knowledge Discovery and Data Mining, pp.156-167. 27-30 April 2009. Bangkok, Thailand. 2009.

[15] A. Narang and S. Bhattacherjee. Parallel Exact Time Series Motif Discovery. Proceedings of the 16th International Euro-Par Conference, pp.304-315. 31 August - 3 September 2010. Ischia, Italy. 2010.

[16] K. Buza and L. Schmidt-Thieme. Motif-Based Classification of Time Series with Bayesian Networks and SVMs. Advances in Data Analysis, Data Handling and Business Intelligence 2010

[17] J. K. Kim and S. Choi. Clustering Sequence Sets for Motif Discovery. Proceedings of The Neural Information Processing Systems, pp.970-978. 10-11 December 2009. Whistler, Canada. 2009.

ประวัติผู้เขียนวิทยานิพนธ์

นายปวัน นันทานิช เกิดเมื่อวันที่ 8 ตุลาคม 2527 สำเร็จการศึกษาในระดับมัธยมศึกษาจากโรงเรียนสาธิตมหาวิทยาลัยศิลปากร วิทยาเขตพระราชวังสนามจันทร์ เข้าศึกษาต่อในระดับอุดมศึกษาที่คณะวิศวกรรมศาสตร์ ภาควิชาวิศวกรรมคอมพิวเตอร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ในปีการศึกษา 2546 และสำเร็จการศึกษาในปีการศึกษา 2549 หลังสำเร็จการศึกษาระดับปริญญาตรี ได้เข้าทำงานในบริษัทเอกชนในสาขาเทคโนโลยีตำแหน่งนักพัฒนาโปรแกรมเป็นเวลา 3 ปี จึงเข้าศึกษาต่อในระดับปริญญามหาบัณฑิตในหลักสูตรวิศวกรรมศาสตรมหาบัณฑิต สาขาวิศวกรรมคอมพิวเตอร์ ที่ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ปีการศึกษา 2553