บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

ทฤษฎีที่เกี่ยวข้อง

1. การจัดกลุ่มข้อมูลสำหรับการทำเหมืองข้อมูล (Data Clustering in Data Mining)

การทำเหมืองข้อมูลสามารถอธิบายได้อย่างง่ายว่าเป็นกระบวนการทำงานอย่างอัตโนมัติสำหรับการค้นหาองค์ความรู้จากข้อมูลที่มีอยู่เป็นจำนวนมาก จากการรวบรวมข้อมูลที่มีอยู่เดิมหรือการค้นหารูปแบบข้อมูลที่ซ่อนไว้ การค้นหาองค์ความรู้จะประกอบด้วย

1. ความเข้าใจในทางธุรกิจและข้อมูล
2. กระบวนการก่อนหน้าที่มีประสิทธิภาพ
3. การทำเหมืองข้อมูลและการทำรายงาน

การทำเหมืองข้อมูลประกอบไปด้วยเทคนิคในการจัดแบ่งประเภท (Classification) การเชื่อมโยงกัน (Association) และ การจัดแบ่งกลุ่ม (Clustering) ซึ่งใช้ในการค้นหาข้อมูลด้วยกฎและรูปแบบที่แตกต่างกัน ในการจัดกลุ่มข้อมูล (Clustering) มีเทคนิคที่นิยมใช้แบ่งออกได้เป็น 2 เทคนิค คือ (1) การแบ่งกลุ่ม (Partitioning) และ (2) การจัดลำดับการเข้าถึงแบบลำดับชั้น (Hierarchical) การแบ่งกลุ่มเป็นอัลกอริทึมที่มีประสิทธิภาพในเชิงเวลาเนื่องจากใช้เวลาในการแบ่งกลุ่มเป็นแบบสมการเชิงเส้นจึงมีความนิยมอย่างแพร่หลาย อัลกอริทึมหนึ่งที่เป็นที่รู้จักดีคือ K-means

1.1 K-means อัลกอริทึม

เป็นอัลกอริทึมที่เข้าใจง่ายและตรงไปตรงมา จะแบ่งข้อมูลออกเป็นกลุ่มตามที่กำหนดไว้ล่วงหน้า (K) จะเริ่มด้วยการสุ่มกำหนดจุดกึ่งกลางของกลุ่มข้อมูล (centroids) จับกลุ่มข้อมูล โดยดูจากความเหมือนกัน และกำหนดจุดกึ่งกลางของกลุ่มข้อมูลที่ได้จัดกลุ่มใหม่และวนทำไปเรื่อย ๆ จนกว่าจะพบเงื่อนไขที่กำหนด เช่น การกำหนดรอบที่แน่นอนของการเวียนเกิด หรือจุดกึ่งกลางที่ได้ไม่มีการเปลี่ยนแปลง สรุปอัลกอริทึมของ K-means ได้ดังนี้

1. ทำการสุ่มเลือกกำหนดค่าให้เป็นเวคเตอร์ของจุดกึ่งกลางของกลุ่มข้อมูลเป็นค่าเริ่มต้นในการแบ่งกลุ่มข้อมูล

2. กำหนดค่าของเวคเตอร์ของแต่ละเอกสารข้อมูลแต่ละตัวให้กับจุดกึ่งกลางที่ใกล้ที่สุด

3. คำนวณซ้ำเพื่อหาเวคเตอร์ของจุดกึ่งกลางที่จัดกลุ่ม ด้วยสูตร

$$c_j = \frac{1}{n_j} \sum_{\forall d_j \in S_j} d_j$$

โดย d_j หมายถึง เวคเตอร์ของเอกสารซึ่งเป็นของกลุ่มข้อมูลที่ถูกแบ่ง S_j

c_j หมายถึง เวคเตอร์ของจุดกึ่งกลางของข้อมูล

n_j หมายถึง จำนวนทั้งหมดของเวคเตอร์ของเอกสารซึ่งเป็นของกลุ่มข้อมูลที่ถูกแบ่ง S_j

4. ทำซ้ำขั้นตอนที่ 2 และ 3 จนกระทั่งไม่สามารถแบ่งกลุ่มได้อีก

ปัญหาอย่างหนึ่งของ K-means อัลกอริทึมคือผลลัพธ์ที่ได้ขึ้นอยู่กับการกำหนดค่าเริ่มต้นให้กับจุดกึ่งกลางของกลุ่มข้อมูลที่ต้องการแบ่ง และการลู่เข้าสู่ local optima

2. **เชาว์ปัญญาเชิงเคลื่อนที่เป็นกลุ่ม** (Swarm Intelligence)

การทำเหมืองข้อมูลกับเชาว์ปัญญาเชิงเคลื่อนที่เป็นกลุ่มอาจดูเหมือนว่าตามปกติมันไม่มีทางเกี่ยวข้องกันได้ อย่างไรก็ตามการศึกษาที่ผ่านได้แสดงให้เห็นว่ามันสามารถที่จะนำมาใช้ร่วมกันได้ในการแก้ไขปัญหาการทำเหมืองข้อมูลจริง เมื่อวิธีการอื่น ๆ นั้นไม่คุ้มค่าและยากที่จะนำมาใช้จริงได้

เชาว์ปัญญาเชิงเคลื่อนที่เป็นกลุ่ม Swarm Intelligence (SI) เกิดจากแนวคิดการจำลองแบบพฤติกรรมของกลุ่มสิ่งมีชีวิตตามธรรมชาติ เช่น อาณาจักรของมด ฝูงนก ฝูงผึ้ง หรือฝูงปลา ในการออกไปหาแหล่งอาหาร โดยปกติ SI จะประกอบไปด้วยกลุ่มประชากรของตัวแทนที่ไม่ซับซ้อนแต่ละตัวแทนจะสื่อสารกันภายในพื้นที่และสื่อสารกับสภาพแวดล้อมได้ แต่ละตัวแทนภายในกลุ่มจะทำหน้าที่ง่าย ๆ เป็นอิสระต่อกันแต่สามารถทำงานร่วมกันได้ทำให้เกิดการทำงานเป็นกลุ่มโดยแต่ละตัวแทนเดี่ยว ๆ ภายในกลุ่มพฤติกรรม โดยรวมของการทำงานเป็นกลุ่มของสิ่งมีชีวิตจะมีคุณลักษณะไม่เป็นเชิงเส้น (nonlinear) และแต่ละตัวแทนเดี่ยว ๆ ก็จะมีพฤติกรรมของตัวเอง เมื่อรวมพฤติกรรมของแต่ละตัวแทนเดี่ยว ๆ นั้นก็จะได้พฤติกรรมโดยรวมของกลุ่มพฤติกรรม ในขณะเดียวกันพฤติกรรมโดยรวมของแต่ละกลุ่มพฤติกรรมก็เป็นตัวกำหนดเงื่อนไขที่ตัวแทนเดี่ยวจะต้องกระทำการกระทำดังกล่าว ซึ่งจะมีผลต่อสภาวะแวดล้อมทำให้พฤติกรรมของตัวแทนเดี่ยวนั้น ๆ เปลี่ยนไปด้วยการกำหนดเงื่อนไขของพฤติกรรมรวมของกลุ่มที่เป็นแบบ

เชิงพื้นที่และเชิงเวลา พฤติกรรมการเคลื่อนที่ของกลุ่มไม่ได้กำหนดด้วยตัวแทนตัวใดตัวหนึ่ง แต่การสื่อสารโต้ตอบระหว่างตัวแทนแต่ละตัวจะเป็นตัวกำหนดพฤติกรรมของกลุ่ม การสื่อสารโต้ตอบระหว่างกันทำให้เกิดการกลั่นกรองและแลกเปลี่ยนประสบการณ์ความรู้เกี่ยวกับสภาพแวดล้อมและเพิ่มสมรรถนะการค้นหาคำตอบที่เหมาะสมที่สุดของปัญหาผลลัพธ์ที่ได้จากการทำงานเป็นกลุ่มคือการจัดการตัวเอง (self-organize)

2.1 ACO อัลกอริทึม

การหาค่าที่เหมาะสมที่สุดด้วยโคโลนีมด Ant Colony Optimization (ACO) ได้นำเสนอแนวคิดเป็นครั้งแรก โดย Dorigo จำลองพฤติกรรมของมด ซึ่งอยู่ภายในอาณาจักรเดียวกันช่วยกันทำหน้าที่หาอาหาร โดยออกจากรังอย่างไม่รู้เส้นทางล่วงหน้าอาศัยการเดินทางแบบสุ่มเดิน และอาศัยฟีโรโมน (pheromone) ซึ่งตกอยู่ตามเส้นทางของมดแต่ละตัวในการค้นหาอาหาร เมื่อมดตัวใดตัวหนึ่งค้นพบอาหารก็จะเดินทางกลับมายังรัง เหมือนเพื่อมาบอกเส้นทางให้กับตัวอื่น ๆ รู้แล้วเดินตามทางที่ตนได้ทิ้งฟีโรโมนไว้ เมื่อมดตัวอื่น ๆ เดินตามทางก็จะยิ่งทำให้ความหนาแน่นของฟีโรโมนในเส้นทางนั้น ๆ เพิ่มมากขึ้น ทำให้มดตัวอื่น ๆ เดินตามมาทางนี้ด้วยกันหมดแล้วมาช่วยกันนำอาหารกลับรังของมัน เมื่อนำการจำลองพฤติกรรมของมด มาช่วยแก้ไขปัญหาที่ต้องการหาค่าที่เหมาะที่สุด Dorigo,Maniezzo และ Colorni (1996) และ ได้นำ ACO อัลกอริทึมนี้กับการแก้ปัญหาการเดินทางของเซลล์แมนไปยังเมืองต่าง ๆ แบบต้องการเดินทางให้สั้นที่สุด และไม่เดินทาง ซ้ำกลับมายังเมืองเดิม ให้เดินทางผ่านแต่ละเมืองเพียงครั้งเดียว โดยจำลองมดด้วยตัวแทน หรือ agent สรุปอัลกอริทึมของ ACO สำหรับปัญหาการเดินทางของเซลล์แมน ได้ดังนี้

1. ทำการสุ่มปริมาณฟีโรโมนเริ่มต้น $\tau_{ij}(0)$ ของแต่ละเส้นทาง (i, j) ระหว่างเมือง i และเมือง j ด้วยค่าสุ่มบวกจำนวนน้อยๆ นั่นคือ $\tau_{ij}(0) \sim U(0, max)$
2. ทำการเริ่มปล่อยตัวแทนมด (agent) ตัวที่ k โดยที่ k $\in$1,...,M ณ เมืองหรือจุดเริ่มต้น
3. กำหนดให้ T^+ เป็นเส้นทางเดินที่สั้นที่สุดและ L^+ เป็นระยะทางของเส้นทางที่สั้นที่สุดนั้น
4. กำหนดช่วงเวลาใน t = 1 ถึง t_{max} ทำซ้ำขั้นตอนต่อไปนี้

4.1 สำหรับตัวแทนมดแต่ละตัว (agent) ทำการสร้างเส้นทาง T_k (t) ของมดตัวที่ k โดยทำการเลือกเมืองถัดไปทั้งหมด N - 1 ครั้ง ด้วยค่าความน่าจะเป็น

$$\Phi_{ij}^{k}(t) = \begin{cases} \frac{\tau_{ij}(t)^{\alpha}}{\tau_{c}(t)} & \text{ถ้า } j \in C_i^k \\ 0 & \text{ถ้า } j \notin C_i^k \end{cases}$$

โดยที่ k แทน ตัวแทน ตัวที่ k ส่วน $T_{ij}(t)$ เป็นปริมาณของฟีโรโมนในเส้นทาง (i, j) ระหว่างเมือง i และเมือง j ค่า α เป็นค่าคงที่ C_i^k เป็นเซตของเมืองที่ตัวแทนตัวที่ k ยังไม่ได้เดินทางไป โดยนับจากเมือง i ส่วน $T_C(t)$ เป็นผลรวมของฟีโรโมนในเส้นทางทั้งหมดของ C_i^k นั่นคือ

$$\tau_c(t) = \sum_{c \in C_i^k} \tau_{ic}(t)^\alpha$$

4.2 คำนวณระยะทางการเดินทาง $L^K(t)$ ของตัวแทน ตัวที่ k

4.3 ถ้าพบเส้นทางที่ดีกว่า ให้ทำการบันทึกค่าแทน T^+ และ L^+

4.4 ทำการปรับค่าฟีโรโมนของแต่ละเส้นทางดังนี้

$$\tau_{ij}(t+1) = (1-\rho)\tau_{ij}(t) + \Delta\tau_{ij}(t)$$

โดยที่ $\Delta\tau_{ij}(t) = \sum_{k=1}^{M} \Delta\tau_{ij}^k(t)$ เป็นค่าผลรวมของฟีโรโมนที่ปล่อยโดยตัวแทนแต่ละตัว ซึ่งคำนวณได้จาก

$$\Delta\tau_{ij}^k(t) = \begin{cases} Q/L^k(t) & \text{ถ้า } (i,j) \in T^k(t) \\ 0 & \text{ถ้า } (i,j) \notin T^k(t) \end{cases}$$

Q เป็นพารามิเตอร์ของระบบ ค่าคงที่ $\rho \in [0, 1]$ เรียกว่าตัวประกอบการลืม (forgetting factor) ที่ซึ่งแทนการจางหายไปของฟีโรโมนเมื่อเวลาผ่านไป ค่าความน่าจะเป็นในการเลือกเส้นทางสามารถปรับปรุงได้โดยการเพิ่มข้อมูลเฉพาะถิ่นของเมืองที่ต้องการไป j จากเมือง i ได้ดังนี้

$$\Phi_{ij}^k(t) = \frac{\tau_{ij}(t)^\alpha \eta_{ij}^\beta}{\sum_{c \in C_i^k} \tau_{ic}(t)^\alpha \eta_{ic}^\beta}$$

โดยที่ α และ β เป็นพารามิเตอร์ของระบบที่ควบคุมน้ำหนักของฟีโรโมนและข้อมูลเฉพาะถิ่น และ

$$\eta_{ij} = \frac{1}{d_{ij}}$$

โดยที่ d_{ij} เป็นระยะทางยุคลิด (Euclidean distance) ระหว่างเมือง i และเมือง j (พิจารณาในระนาบสองมิติ) สังเกตว่าค่าของ Φ_{ij}^k อาจจะแตกต่างกันสำหรับตัวแทนแต่ละตัว ณ เมืองเดียวกันได้ อันเนื่องมาจากตัวแทนแต่ละตัวอาจจะมาถึงเมืองนั้นๆ ด้วยเส้นทางที่แตกต่างกัน อัลกอริทึมแบบจำลองพฤติกรรมของมดข้างต้น มีพารามิเตอร์ของระบบที่ต้องพิจารณาค่อนข้างมาก

พารามิเตอร์ที่ถือว่าสำคัญที่สุด คือ ตัวประกอบการลืมสำหรับการจางหายไปของฟีโรโมน และจำนวนของตัวแทน M ในระบบ ในลักษณะเดียวกันกับอัลกอริทึมประเภทอื่นๆ จำนวนตัวแทนที่มากเกินไปจะใช้เวลาในการคำนวณมากไปด้วย รวมไปถึงพฤติกรรมการลู่เข้าสู่คำตอบที่ไม่ใช่คำตอบที่เหมาะที่สุดอย่างเร็วเกินไป ในทางตรงกันข้าม จำนวนตัวแทนที่น้อยเกินไปทำให้การทำงานร่วมกันของตัวแทนไม่เพียงพอต่อการค้นหาคำตอบที่เหมาะที่สุด นอกเหนือไปจากพารามิเตอร์เหล่านี้แล้ว พารามิเตอร์ α และ β ยังเป็นสิ่งที่ต้องคำนึงถึง ถ้า $\beta = 0$ แสดงว่าระบบใช้ข้อมูลจากฟีโรโมนเพียงอย่างเดียว ซึ่งอาจจะนำไปสู่คำตอบที่ไม่ใช่ค่าที่เหมาะที่สุดได้ ถ้า $\alpha = 0$ แสดงว่าระบบไม่ได้ใช้ข้อมูลของฟีโรโมนเลย การค้นหาของตัวแทนจะกลายเป็นเพียงการค้นหาคำตอบแบบตะกละตะกลาม

2.2 PSO **อัลกอริทึม**

การหาค่าที่เหมาะที่สุดด้วยการเคลื่อนที่ของกลุ่มอนุภาค Particle Swarm Optimization (PSO) อัลกอริทึมนี้ออกแบบและนำเสนอโดย Kennedy และ Eberhart (1995) PSO คือ การจำลองพฤติกรรมทางสังคมของฝูงนก ฝูงผึ้งหรือฝูงปลา มาเป็นอัลกอริทึมในการค้นหาโดยอาศัยความน่าจะเป็น โดยการจำลองพฤติกรรมของการออกไปหาอาหาร เช่น การออกไปหาอาหารของนก โดยนกในฝูงจะช่วยกันออกไปหาอาหาร โดยที่แต่ละตัวจะไม่รู้ล่วงหน้าถึงสถานที่ที่มีอาหารอยู่ที่ใด แต่ละครั้งของการบินออกไปหาอาหารมันจะเรียนรู้ด้วยการติดต่อสื่อสารกันภายในฝูงถึงระยะทางในการออกไปหาอาหาร ดังนั้นเมื่อตัวใดตัวหนึ่งพบอาหารจะเกิดการเปรียบเทียบเพื่อไปยังแหล่งอาหารที่อยู่ใกล้ที่สุดที่ค้นพบจากพฤติกรรมดังกล่าวของฝูงนก PSO ได้นำมาใช้ในการหาค่าที่เหมาะที่สุดในการแก้ปัญหา ใน PSO นกแต่ละตัวถูกเรียกใหม่ด้วยคำว่า อนุภาค (particle) ทุก ๆ อนุภาคจะมี fitness values ซึ่งจะถูกประเมินผลด้วย fitness function และมี velocities ซึ่งคือเส้นทางการบินหรือเคลื่อนที่ของอนุภาคแต่ละอนุภาคก็จะเคลื่อนที่ตามตัวที่ได้ค่าที่เหมาะที่สุดในขณะนั้น ในการเริ่มต้นกระบวนการ PSO จะสุ่มสร้างกลุ่มของอนุภาค แล้วเริ่มตามกระบวนการค้นหาในการหาค่าที่เหมาะสมที่สุดในการปรับปรุงในแต่ละรุ่นหรือช่วงอายุของอนุภาค ระหว่างการเวียนเกิด (iteration) ในทุก ๆ รอบ อนุภาคแต่ละตัวจะทำการปรับปรุงค่าที่ดีที่สุดสองค่า ค่าแรกคือ ตำแหน่งที่ดีที่สุดในเส้นทางของตัวมันเอง ซึ่งเรียกว่า *pbest* ค่าที่สอง คือค่าที่เหมาะสมที่สุดที่ได้จากการเคลื่อนที่ของกลุ่มอนุภาค ซึ่งเกิดจากการค่าความน่าจะเป็นที่ได้ในแต่ละอนุภาคในกลุ่ม ซึ่งเรียกว่า *gbest* แต่กระนั้น PSO ก็ยังมีปัญหาที่เกิดตามมาคือการลู่เข้าสู่ local optimal (ค่าที่ดีที่สุดในท้องถิ่น) ในการที่จะปรับปรุงประสิทธิภาพของ PSO ได้มีงานวิจัยที่นำเสนอต่อมาอีกมากมาย บางอัลกอริทึมที่นำเสนอทำการแก้ไขปัญหานี้ด้วย

การผสมผสานอัลกอริทึมอื่นเข้าไป แม้ว่านักวิจัยทั้งหลายจะถือว่าอัลกอริทึม ของตนดีกว่า PSO แบบดังเดิม แต่ก็เกิดผลกับการทำงานจริงในเรื่องของการใช้เวลาในการทำงานของอัลกอริทึม รวมถึงอัลกอริทึมนั้นเข้าใจยาก ณ ปัจจุบันจึงยังไม่มีรูปแบบทฤษฎีใดที่รองรับหรือสนับสนุนการปรับปรุงแก้ไข PSO ที่นักวิจัยได้นำเสนอ สรุปอัลกอริทึมของ PSO ได้ดังนี้

1. ทำการสุ่มค่าเวกเตอร์ตำแหน่งและความเร็วของแต่ละอนุภาคในกลุ่ม เวกเตอร์ตำแหน่งของอนุภาคมีมิติเท่ากับ N ซึ่งเป็นขนาดของตัวแปรในปัญหาที่ต้องการค้นหา ดังนั้น ตำแหน่งของอนุภาคในรูปเวกเตอร์ $\vec{p}_i(t)$จะมีขนาด N และค่าสุ่มความเร็วของแต่ละอนุภาคมีขนาดเท่ากับ N เรียกว่าเวกเตอร์ความเร็ว (velocity vector) โดยกำหนดให้ $\vec{v}_i$ แทนเวกเตอร์ความเร็ว แต่ละองค์ประกอบของเวกเตอร์ความเร็วจะเป็นค่าความเร็วของแต่ละตัวแปรในอนุภาคนั้นเองดังนั้นตำแหน่งของอนุภาค P_i จะเปลี่ยนแปลงไปด้วยการบวกเวกเตอร์ตำแหน่งเข้ากับเวกเตอร์ความเร็วดังนี้

$$\vec{p}_i(t) = \vec{p}_i(t-1) + \vec{v}_i(t)$$

2. ประเมินค่าความเหมาะสม $\mathcal{F}$ ของแต่ละอนุภาคซึ่งการประเมินค่าดังกล่าวจะขึ้นอยู่กับแต่ละปัญหาค่าความเหมาะสมที่ได้จากการประเมินจะถูกพิจารณาในสองขั้นตอนดังนี้

2.1 ถ้าค่าความเหมาะสมของอนุภาค Pi มีค่าดีกว่าค่าความเหมาะสมที่ดีที่สุดของทั้งกลุ่มอนุภาคให้ทำการบันทึกเวกเตอร์ตำแหน่งของอนุภาคนี้ไว้โดยเรียกว่า ค่าความเหมาะสม ที่ดีที่สุดแบบวงกว้าง (global best fitness) หรือ *gbest* กล่าวคือถ้า $\mathcal{F}\left(\vec{p}_i(t)\right) < gbest$ (กรณีหาค่าที่น้อยที่สุด) ให้ทำการบันทึกค่าความเหมาะสมของระบบและค่าเวกเตอร์ตำแหน่งของอนุภาคนั้น ๆ ไว้ดังต่อไปนี้

- $gbest = \mathcal{F}\left(\vec{p}_i(t)\right)$
- $\vec{p}_{gbest} = \vec{p}_i(t)$

2.2 ถ้าค่าความเหมาะสมของอนุภาค P_i มีค่าดีกว่าค่าความเหมาะสมที่ดีที่สุดของอนุภาคนั้น ๆ ซึ่งเรียกว่า $pbest_i$ (p มาจาก personal) กล่าวคือ $\mathcal{F}\left(\vec{p}_i(t)\right) < pbest_i$ ให้ทำการบันทึกค่าความเหมาะสมและเวกเตอร์ตำแหน่งของอนุภาคไว้ใน $pbest_i$ ดังนี้

- $pbest_i = \mathcal{F}\left(\vec{p}_i(t)\right)$
- $\vec{p}_{pbest_i} = \vec{p}_i(t)$

2.3 ทำการปรับค่าความเร็วของอนุภาค P_i ดังนี้

$$\vec{v}_i(t) = \vec{v}_i(t-1) + \rho_p \left[\vec{p}_{pbest_i} - \vec{p}_i(t)\right] + \rho_g \left[\vec{p}_{gbest} - \vec{p}_i(t)\right]$$

โดยที่ ρ_p และ ρ_g เป็นตัวแปรสุ่มเทอมที่สองของสมการข้างต้นเรียกว่าองค์ประกอบเชิงปริชาน (cognitive component) และเทอมสุดท้ายเรียกว่าองค์ประกอบทางสังคม (social component)

2.4 ทำการปรับค่าเวกเตอร์ตำแหน่งของอนุภาค Pi ดังนี้

$$\vec{p}_i(t) = \vec{p}_i(t-1) + \vec{v}_i(t)$$

2.5 ทำการปรับค่าตัวแปรเวลา $t = t + 1$ และดำเนินขั้นตอนทั้งหมดกับอนุภาคถัดไปในกลุ่มจนครบทุกอนุภาค

2.6 วนรอบการทำงานทั้งหมดจนกระทั่งมีการลู่เข้าสู่คำตอบของอนุภาคที่ดีที่สุดในกลุ่มหรือตามเงื่อนไขหยุดที่กำหนดไว้

งานวิจัยที่เกี่ยวข้อง

จากอัลกอริทึม PSO แบบดั้งเดิม ได้มีงานวิจัยเกิดขึ้นตามมาอีกมากมาย เพื่อเพิ่มประสิทธิภาพให้กับ PSO ซึ่งอาจแบ่งได้เป็นสองแนวทาง กล่าวคือ

1. งานวิจัยที่เสนอการปรับปรุงประสิทธิภาพของ PSO ด้วยการปรับปรุงที่อัลกอริทึมเอง หรือหาค่าพารามิเตอร์ที่เหมาะสมสำหรับปัญหาหรือข้อมูลนั้น ๆ ซึ่งผลงานวิจัยเชิงการทดลองก็สรุปได้ว่ามีประสิทธิภาพดีขึ้น Ahmadi, Karrayi, และ Karnel (2008) นำเสนอการทำงานร่วมกันของกลุ่มเคลื่อนที่เพื่อแบ่งกลุ่มข้อมูล เปรียบเทียบกับหลายอัลกอริทึมแล้วได้ผลดี และค้นพบว่าค่าความน่าจะเป็นของการทำงานร่วมกันของกลุ่มเคลื่อนที่ มีค่ามากกว่ากลุ่มเคลื่อนที่เดียว นำมาใช้เป็นเทคนิคในการค้นหารูปแบบของข้อมูลได้ Jarboui, Cheikh, Siarry และ Rebai (2007) นำเสนอแนวคิดการควบรวมของการเคลื่อนที่เพื่อหาค่าเหมาะที่สุดมาทำการจัดกลุ่มข้อมูลเปรียบเทียบผลกับอัลกอริทึม GA ได้ผลดีกว่าในบางกลุ่มข้อมูลที่นำมาทดลอง ขึ้นอยู่กับปัจจัยอื่น ๆ เช่น การกำหนดค่าเริ่มต้นให้กับพารามิเตอร์ Mahamed, Omran, Engelbrecht และ Salman (2005) นำเอาอัลกอริทึม PSO มาทำการแบ่งประเภทรูปโดยการกำหนดจำนวนของกลุ่มข้อมูลที่ดีที่สุดในรูปภาพที่นำมาทดสอบ เพื่อบ่งชี้ความแตกต่างของรูปภาพหนึ่ง ๆ ได้

2. งานวิจัยที่เสนอการปรับปรุงประสิทธิภาพของ PSO ด้วยการรวมหลาย ๆ เทคนิคเข้าด้วยกัน อาจทำให้เสียเวลาในการประมวลผลเพิ่มขึ้น ก็ต้องแลกกับการได้คำตอบที่ถูกต้อง

หรือผิดพลาดน้อยลงสำหรับการแก้ปัญหานั้น ๆ Yang, Sun และ Zhang (2009) นำเสนอการจัดกลุ่มข้อมูลที่รวม 2 เทคนิค คือ k-harmonic means และ PSO เพื่อช่วยหลีกเลี่ยงปัญหาของการเกิด local optima ของทั้งสองอัลกอริทึมที่นำมารวมกัน แต่กระนั้นอัลกอริทึมใหม่ที่นำเสนอก็ใช้เวลามากกว่าเมื่อเทียบกับอัลกอริทึมเดิมทั้งสองตัว แต่ก็ช่วยชลอการลู่เข้าสู่ local optima ให้เกิดขึ้นได้อยาก อีกหนึ่งงานวิจัยที่ใช้เทคนิคควบรวมอัลกอริทึมหลาย ๆ ตัว เพื่อการจัดกลุ่มข้อมูล Kao, Y., Zahara, และ Kao, I (2007) นำเอา k-means อัลกอริทึม, การค้นหาปกติแบบ Neider-Mead และ PSO มาทำการจัดกลุ่มข้อมูลได้ผลการทดลองที่มีอัตราการผิดพลาดน้อย แต่ใช้เวลามากขึ้นเมื่อเทียบกับอัลกอริทึมแต่ละตัวที่นำมาใช้ควบรวม ในการวิจัยเกี่ยวกับการจัดกลุ่มข้อมูล Sandra และ Leandro นำเอา PSO มาใช้ในการจัดกลุ่มข้อมูลได้ผลการวิจัยดีกว่า k-means อัลกอริทึม เมื่อมาทดลองกับชุดข้อมูลของมะเร็งปอด แต่ไม่ได้ผลดีกับทุก ๆ ข้อมูลเนื่องจากการกำหนดค่าเริ่มต้นในกลุ่มอนุภาคมีความเป็นไปได้มากมายและกำหนดระยะเวลาในการค้นหาคำตอบสำหรับอัลกอริทึมอาจจะน้อยไปทำให้การค้นหาจุดกึ่งกลางของข้อมูลที่ต้องการไม่ถูกต้องหรือการเลือกฟังก์ชั่นที่ทดสอบหนึ่ง ๆ อาจไม่เหมาะสมได้กับทุก ๆ ชุดข้อมูล

ในความเป็นจริงแล้วการจัดกลุ่มข้อมูลก็ถูกจัดว่า เป็นหนึ่งในปัญหาที่ต้องการค้นหาค่าที่เหมาะที่สุด ในการค้นหาตัวแทนของข้อมูลในแต่ละกลุ่มให้ชัดเจนถูกต้อง แนวทางหนึ่งที่ซึ่งกำลังเป็นหัวข้อที่น่าสนใจในการค้นหาค่าที่เหมาะที่สุดสำหรับการจัดกลุ่มข้อมูล คือ การนำเทคนิคหรืออัลกอริทึมของการหาค่าที่เหมาะที่สุดด้วยการเคลื่อนที่ของกลุ่มอนุภาค มาใช้เพื่อการแก้ไขปัญหา ซึ่งด้วยความสามารถของอัลกอริทึมของ PSO หรือการนำเอาอัลกอริทึมใหม่ที่ได้ปรับปรุงหรือเพิ่มเติมขบวนการให้ PSO แบบดั้งเดิมดีขึ้น มาทำการประยุกต์กับการจัดกลุ่มข้อมูล และวิจัยเชิงการทดลองกับกลุ่มข้อมูลที่มีมาตรฐานเพื่อเปรียบเทียบผลที่ได้กับอัลกอริทึมในการจัดกลุ่มแบบเดิมของการทำเหมืองข้อมูล

1. EPSO สำหรับการจัดกลุ่มข้อมูล

Alam,Dobbie และ Riddle (2008) ได้นำเสนอการเพิ่มประสิทธิภาพในการจัดกลุ่มข้อมูลด้วยการอาศัยพื้นฐานของ PSO แบบดั้งเดิม ด้วยการนำการประเมินผลแบ่งตามรุ่นหรือช่วงอายุสำหรับการหาค่าที่เหมาะสมที่สุดด้วยการเคลื่อนที่ของกลุ่มอนุภาค เกิดจากการรวมกันของสองเทคนิคคือ การจัดการตนเองของกลุ่มเคลื่อนที่ (Self Organization of Swarm) และวิวัฒนาการของอนุภาคแบ่งเป็นรุ่น ๆ เพื่อให้ได้รุ่นสุดท้ายที่ค้นพบค่าที่เหมาะที่สุดในการจัดกลุ่ม

ข้อมูลโดยในแต่ขบวนการของการแบ่งรุ่น (generation) อนุภาคที่แข็งแกร่งกว่าจะคงอยู่ อนุภาคที่อ่อนแอจะถูกกำจัดโดยควบรวมเข้ามาเป็นตัวเดียวกันกับอนุภาคที่แข็งแกร่งกว่าที่อยู่ใกล้ที่สุด สรุปอัลกอริทึมของ ESPO ได้ดังนี้

1. กำหนดค่าเริ่มต้นให้แต่ละอนุภาค
    1.1 กำหนดค่าเริ่มต้นให้แก่
        $V_i(t)$ หมายถึง เส้นทางปัจจุบัน
        $X_i(t)$ หมายถึง ตำแหน่งของอนุภาคแต่ละตัว
        V_{max} หมายถึง ค่าสูงสุดของเส้นทางที่กำหนดไว้ เท่ากับ 0.069
        ω หมายถึง น้ำหนักเริ่มต้น เท่ากับ 0.99
        q_1 หมายถึง ค่าคงที่ของน้ำหนักองค์ประกอบเชิงปริชาน เท่ากับ 0.005
        q_2 หมายถึง ค่าคงที่ของน้ำหนักขององค์ประกอบทางสังคม เท่ากับ 0
    1.2. กำหนดค่าเริ่มต้นของขนาดกลุ่มเคลื่อนที่ เท่ากับ 5 และการกำหนดรุ่นหรือแบ่งรุ่น เท่ากับ 10
    1.3. กำหนดค่าเริ่มต้นให้อนุภาคเป็นข้อมูลที่นำเข้า เท่ากับ 10
2. การเวียนเกิดของรุ่น (generation)
    2.1 การเวียนเกิดของกลุ่มเคลื่อนที่
    2.2 หาผู้ชนะของเส้นทางข้อมูลทั้งหลาย (data vectors)
    2.3 ปรับค่าของเส้นทาง (velocity) และตำแหน่ง (position)
3. ประเมินความแข็งแกร่งของกลุ่มเคลื่อนที่
    3.1. การเวียนเกิดของรุ่น
    3.2. ควบรวมอนุภาคที่อ่อนแอ
    3.3. คำนวณหาตำแหน่งทั้งหมดใหม่
4. ออก ด้วยครบกำหนดตัวเลขของรุ่น เท่ากับ 100 หรือพบข้อกำหนดให้หยุด

สำหรับงานวิจัยนี้นักวิจัยได้ทำการเปรียบเทียบผลกับอัลกอริทึมในการจัดกลุ่มข้อมูลคือ อัลกอริทึม K-means และ การแบ่งกลุ่มข้อมูลด้วย PSO แบบดั้งเดิม โดยทำการจัดกลุ่มข้อมูลตัวเดียวกัน คือ RUSPINI DATA SET ได้ผลการวิจัยเชิงการทดลองดีกว่าทั้งสองตัวดังตารางนี้

ตารางที่ 1

ตารางเปรียบเทียบผลการจัดกลุ่มข้อมูลของ RUSPINI DATA SET ด้วยอัลกอริทึม K-means, PSO-Clustering & EPSO-Clustering

Method	Number Of Clusters	Cluster#	Number of Data Vectors	Distance from centroid	Avg. Intra-Cluster distance
K-means	4	1	20	12.5297	11.3967
		2	23	10.3833	
		3	17	13.6997	
		4	15	8.9739	
PSO-Clustering	4	1	20	12.5959	11.4296
		2	23	10.4051	
		3	17	13.7591	
		4	15	8.9583	
EPSO-Clustering	4	1	20	12.5377	11.3940
		2	23	10.319	
		3	17	13.7585	
		4	15	8.9450	

ที่มา: “An evolutionary particle swarm optimization algorithm for data clustering,” by Alam, S., Dobbie, G., & Riddle, P. (2008), 2008 IEEE Swarm intelligence symposium, St.,Louis MO USA., September 21-23,2008. IEEE.

2. MSA-PSO

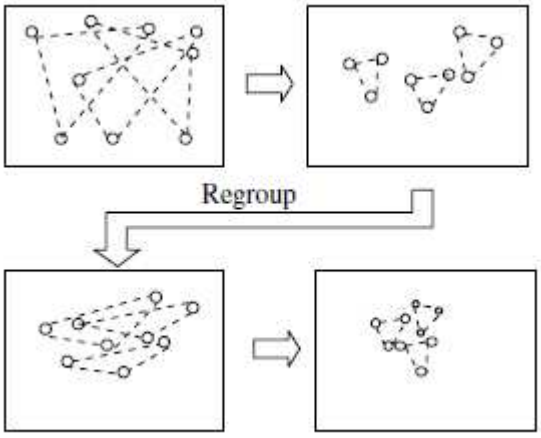
Jiang,Huang และ Chen (2009) ได้นำเสนอการวิธีการแก้ปัญหาฟังก์ชั่นต่อเนื่องแบบไม่คงที่ (Dynamic continuous functions) ด้วยการปรับปรุง กลุ่มเคลื่อนที่แบบหลายกลุ่ม

(Multi-Swarm) โดยการเพิ่มกระบวนการใหม่คือการเปลี่ยนกลุ่มใหม่เข้าไปในอัลกอริทึมของ Multi-Swarm การสุ่มการเปลี่ยนกลุ่มนี้ทำให้อนุภาคมีการเปลี่ยนโครงสร้างของเพื่อนบ้านแบบไม่คงที่ เพื่อหลุดพ้นจากปัญหา local optima และยังทำให้โครงสร้างของกลุ่มเคลื่อนที่เป็นแบบไม่คงที่อีกด้วย

สมมุติว่ามีกลุ่มทั้งหมดสามกลุ่ม และแต่ละกลุ่มมีสามอนุภาค อย่างแรกอนุภาคทั้งเก้าตัวจะถูกแบ่งออกเป็นกลุ่มเคลื่อนที่สามกลุ่มแบบสุ่ม หลังจากนั้นแต่ละกลุ่มจะใช้อนุภาคของตนในการค้นหาหนทางแก้ปัญหาที่ดีขึ้น ในระหว่างนี้อาจจะมีการลู่เข้าสู่ local optima เมื่อเกิดการเปลี่ยนกลุ่มไปอยู่กลุ่มใหม่และเริ่มต้นค้นหาใหม่ กระบวนการนี้จะดำเนินไปเรื่อย ๆ จนพบเงื่อนไขให้หยุด ด้วยการสุ่มการเปลี่ยนกลุ่มตามตารางเวลา อนุภาคแต่ละตัวในแต่ละกลุ่มจะถูกกำหนดค่าให้ใหม่ ดังนั้นพื้นที่ในการค้นหาของอนุภาคแต่ละตัวจะมีขนาดใหญ่ขึ้นและการแก้ไขปัญหาที่ดีอาจจะถูกค้นพบด้วยกลุ่มใหม่

ภาพที่ 1

ตัวอย่างการเปลี่ยนกลุ่มของ MSA-PSO



โอเปอร์เรเตอร์ของการเร่ง

ความหลากหลายที่มากและเส้นทางที่มาบรรจบกันได้เร็วขึ้น ทั้งสองอย่างนี้จะต้องเป็นปัญหาที่ถูกเปรียบเทียบกันเสมอ ถ้าเราได้ความหลากหลายที่มากของเส้นทางก็จะขาดความรวดเร็วในการค้นหาทางที่เร็วที่สุด ทางที่จะช่วยให้จุดอ่อนนี้ดีขึ้น คือ การเพิ่มความสามารถ

ในการค้นหาแบบท้องถิ่น (local search) ที่ดีขึ้น และการค้นหาแบบท้องถิ่นนี้ถูกเพิ่มเข้าไปใน MSA-PSO

โอเปอร์เรเตอร์ของการเร่งเป็นค่าความน่าจะเป็นที่ใช้ควบคุมทิศทางของการเคลื่อนที่ด้วยพารามิเตอร์ควบคุมกระบวนการ แนวคิดพื้นฐานก็คือ จะทำการปรับปรุงอนุภาคในอัลกอริทึมคำนวณทุก ๆ ครั้งที่อนุภาคเปลี่ยนค่า fitness value ซึ่งเป็นฟังก์ชั่นเป้าหมาย และบันทึกค่าการเปลี่ยนแปลงของตัวแปรและฟังก์ชั่นเป้าหมาย และประเมินว่าอนุภาคจะเก็บค่าไว้หรือจะข้ามไปสำหรับเส้นทางนั้น ๆ ท้ายสุด ทำการ Alopex operation ซึ่งจะได้ค่าที่คลาดเคลื่อน (noise) ใหม่เฉพาะตัว สำหรับการเร่งการทำงานของอัลกอริทึมในการบันทึกการเปลี่ยนแปลงของฟังก์ชั่นเป้าหมาย กระบวนการเร่งมีดังนี้

$$x_i(n) = x_i(n-1) + \delta_i(n) \tag{1}$$

$$\delta_i(n) = \begin{cases} \delta_i, p_i(n) = p_c(n) \cap p_i(n) \geq r_1 \\ r\delta_i, p_i(n) = p'_c(n) \cap p_i(n) \geq r_1 \\ -\delta_i, 1 - p_i(n) = p_c(n) \cap p_i(n) < r_1 \\ -r\delta_i, 1 - p_i(n) = p'_c(n) \cap p_i(n) < r_1 \end{cases} \tag{2}$$

$$p_i(n) = \frac{1}{1 + e^{\pm \Delta_i(n)/T}} \tag{3}$$

$$\Delta_i(n) = \left[x_i(n-1) - x_i(n-2)\right] \times \left[f(n-1) - f(n-2)\right] \tag{4}$$

โดยที่ $f(n)$ หมายถึง ฟังก์ชั่นเป้าหมายของสำหรับค่า n แต่ละตัว

$x_i(n)$ หมายถึง ฟังก์ชั่นเป้าหมายของตัวแปร i ในแต่ละ n รอบ

$\delta_i(n)$ หมายถึง ตัวแปรเชิงสุ่มของการเคลื่อนที่

$p_i(n)$ หมายถึง เส้นทางที่ค่าความน่าจะเป็นของการเคลื่อนที่เพิ่มขึ้น

r_1 หมายถึง ตัวเลขที่สุ่มขึ้น

$p_c(n)$ หมายถึง ค่าความน่าจะเป็นที่เป็นบวก

$p_c^{'}(n)$ หมายถึง ค่าความน่าจะเป็นที่เป็นลบ

The flowchart of MSA-PSO is:
m: Each swarm's population size
n: Swarm' number
R: Regrouping period
L: Accelerating period
L-FEs: Max FEs during accelerating period
Max-FEs: Max fitness evaluations, stop criterion
Initialize m*n particles (position and velocity)
Divide the population into n swarms randomly, with m particles in each swarm.
FEs=O; gen=0
While FEs < 0.95 *Max-FEs
gen=gen+1;
For i=l:m*n
Find lbesti
For d=l:D
If rand<0.5
Vid= w* Vid + c1 * randlid * (pbestid -Xid)
+c2 * rand2id * (lbestid-Xid)
$V_i^d = \min(\max(V_i^d, -V_{max}^d), V_{max}^d)$
$X_i^d = X_i^d + V_i^d$
Else
$X_i^d = pbest_i^d$
End
End
If $X_i \in [X_{min}, X_{max}]^D$
Calculate the fitness value
FEs= FEs+1
Update pbest
End
End

If mod (gen, L)==0,

Sort the lbest according to their fitness value and refine the first 0.25n best lbest using accelerating operators.
FEs = FEs + 0.25n* L -FEs
Update the corresponding pbest
End
If mod (gen, R)==0,
Regroup the swarms randomly,
End
End
Refine the best solution achieved so far using accelerating operators.

ตารางที่ 2

ตารางเปรียบเทียบข้อดีข้อเสียของงานวิจัย

งานวิจัย	เทคนิค	ข้อดี	ข้อเสีย
Applying Multi-Swarm Accelerating Particle Swarm Optimization to Dynamic Continous Functions	MSA-PSO	1. มีโครงสร้างที่แบบไม่คงที่เพื่อรองรับข้อมูลแบบไม่คงที่ 2. มีขนาดของกลุ่มอนุภาคที่เล็ก และใช้ทรัพยากรน้อยในการทำงาน 3. อัลกอริทึมเข้าใจง่าย	1. ยังไม่มีการนำมาใช้ในการจัดกลุ่มด้วยการทดสอบกับข้อมูลจริง 2. ต้องกำหนดค่าเริ่มต้นที่ใช้ในอัลกอริทึมด้วยผู้ใช้เอง
An Evolutionary Particle Swarm Optimization Algorithm for Data Clustering	EPSO-Clustering	1. อัลกอริทึมเข้าใจง่าย 2. มีประสิทธิภาพที่ดีกว่า K-means และ PSO ในการจัดกลุ่มข้อมูล 3. หลักวิวัฒนาการทำให้การทำงานเร็วขึ้นเรื่อย ๆ ในรอบการเวียนเกิดถัดไป	1. ยังไม่มีการทดสอบกับข้อมูลชนิดอื่นนอกจากที่ระบุในงานวิจัยเพียงชนิดเดียว 2. ต้องกำหนดค่าเริ่มต้นที่ใช้ในอัลกอริทึมด้วยผู้ใช้เอง 3. ต้องเพิ่มเงื่อนไขการตรวจสอบความแข็งแกร่งของกลุ่มอนุภาคในแต่ละรุ่น 4. ขาดการวิเคราะห์ข้อมูลที่ไม่เข้าพวก