

โครงการเลขที่ วศ.คพ. S006-2/65/2566

เรื่อง

ระบบสนับสนุนการตัดสินใจซื้อขายสินทรัพย์ด้วยฟัซซีโลจิก

โดย

ธนต์ถ์ ตั้งอั้น รหัส 630610737
ธนวัฒน์ บำเพ็ญพันธุ์ รหัส 630610736

รายงานนี้เป็นส่วนหนึ่งของวิชาสำรวจเพื่อโครงการ
ตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเชียงใหม่
ปีการศึกษา 2566

PROJECT No. CPE S006-2/65/2566

Fuzzy Logic in Market Trading Decision Support System

Tanat Tangun 630610737
Thanawat Bumpengpun 630610736

**A Report Submitted in Partial Fulfillment of Project Survey Course
as Required by the Degree of Bachelor of Engineering
Department of Computer Engineering
Faculty of Engineering
Chiang Mai University
2023**

หัวข้อโครงการ : ระบบสนับสนุนการตัดสินใจซื้อขายสินทรัพย์ด้วยฟัซซีโลจิก
: Fuzzy Logic in Market Trading Decision Support System
โดย : ธนัตถ์ ตั้งอั้น รหัส 630610737
: ธนวัฒน์ บำเพ็ญพันธุ์ รหัส 630610736
ภาควิชา : วิศวกรรมคอมพิวเตอร์
อาจารย์ที่ปรึกษา : รศ.ดร. ศันสนีย์ เอื้อพันธุ์วิริยะกุล
ปริญญา : วิศวกรรมศาสตรบัณฑิต
สาขา : วิศวกรรมคอมพิวเตอร์
ปีการศึกษา : 2566

ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเชียงใหม่ ได้อนุมัติให้โครงการนี้เป็นส่วน-
หนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต (สาขาวิศวกรรมคอมพิวเตอร์)

..... หัวหน้าภาควิชาวิศวกรรมคอมพิวเตอร์
(รศ.ดร. สันติ พิทักษ์กัจจนกุล)

คณะกรรมการสอบโครงการ

..... ประธานกรรมการ
(รศ.ดร. ศันสนีย์ เอื้อพันธุ์วิริยะกุล)

..... กรรมการ
(ผศ.ดร. เกษมสิทธิ์ ตียพันธ์)

..... กรรมการ
(รศ.ดร. นิพนธ์ อีระอำพน)

หัวข้อโครงการ : ระบบสนับสนุนการตัดสินใจซื้อขายสินทรัพย์ด้วยฟัซซีโลจิก
: Fuzzy Logic in Market Trading Decision Support System
โดย : ธนัตถ์ ตั้งอั้น รหัส 630610737
: ธนวัฒน์ บำเพ็ญพันธุ์ รหัส 630610736
ภาควิชา : วิศวกรรมคอมพิวเตอร์
อาจารย์ที่ปรึกษา : รศ.ดร. ศันสนีย์ เอื้อพันธุ์วิริยะกุล
ปริญญา : วิศวกรรมศาสตรบัณฑิต
สาขา : วิศวกรรมคอมพิวเตอร์
ปีการศึกษา : 2566

บทคัดย่อ

ในการวิเคราะห์ทางเทคนิค มีการใช้อินดิเคเตอร์ทางเทคนิคและปัจจัยอื่นๆมาช่วยในการตัดสินใจ ซึ่งหลายๆอย่างก็มีการตีความหมายด้วยเกณฑ์ที่ไม่สามารถรับความไม่แน่นอนและความผันผวนของตลาดได้ เช่น ค่าคงที่ เป็นต้น และถ้าเราใช้อินดิเคเตอร์ทางเทคนิคหลายๆ อันด้วยกันแล้วการตีความหมายแต่ละอย่างพร้อมๆกันก็เป็นเรื่องที่เราทำได้ยาก ดังนั้นทางผู้จัดจึงสร้างระบบเพื่อช่วยนักลงทุนในการเทรดโดยนำอินดิเคเตอร์ทางเทคนิคและปัจจัยอื่นๆ ของผู้ใช้งานที่ใช้ในการวิเคราะห์การซื้อ และการขายมาสร้างอินดิเคเตอร์ตัวใหม่ที่ช่วยตัดสินใจโดยใช้ Fuzzy logic ซึ่งต่างจากอินดิเคเตอร์ทางเทคนิคแบบดั้งเดิม เนื่องจากสามารถเอามุมมองการวิเคราะห์ส่วนตัวของผู้ใช้งานใส่เข้าไปในอินดิเคเตอร์ตัวนี้ได้ โดยอินดิเคเตอร์ตัวนี้จะรับข้อมูลอย่างเช่น RSI, MA, การทำกำไรของสินทรัพย์, ความผันผวนของตลาด และข้อมูลอื่นๆ ที่ผู้ใช้งานอาจจะต้องการ ในขณะที่เราได้พูดถึงคือสัญญาณการซื้อ และการขาย หรือสัญญาณวิเคราะห์อื่นๆ ที่ผู้ใช้งานต้องการสร้างขึ้น ด้วยวิธีดังกล่าวอินดิเคเตอร์ของเราจะสามารถช่วยนักลงทุนในการจัดการกับข้อมูลหลายๆปัจจัยที่ผู้ใช้งานใช้ในการวิเคราะห์ ออกมาเป็นสัญญาณใหม่เพียง 1 หรือ 2 สัญญาณที่เข้าใจง่าย เพื่อใช้ในการช่วยตัดสินใจ เราจะสร้างเว็บแอปพลิเคชันจากไอเดียดังกล่าวข้างต้น แล้วเผยแพร่เพื่อเก็บผลตอบแทนจากผู้ใช้งาน

สารบัญ

บทคัดย่อ	ข
สารบัญ	ค
สารบัญรูป	ง
สารบัญตาราง	จ
1 บทนำ	1
1.1 ที่มาของโครงการ	1
1.2 วัตถุประสงค์ของโครงการ	1
1.3 ขอบเขตของโครงการ	1
1.4 ประโยชน์ที่ได้รับ	2
1.5 เทคโนโลยีและเครื่องมือที่ใช้	2
1.6 แผนการดำเนินงาน	2
1.7 บทบาทและความรับผิดชอบ	2
1.8 ผลกระทบด้านสังคม สุขภาพ ความปลอดภัย กฎหมาย และวัฒนธรรม	3
2 ทฤษฎีที่เกี่ยวข้อง	4
2.1 ฟัซซีลอจิก (Fuzzy Logic)	4
2.1.1 ฟัซซีเซต (Fuzzy Set)	4
2.1.2 ระบบประมวลผลฟัซซีลอจิก (Fuzzy Logic System)	4
2.2 การหาค่าที่เหมาะสมที่สุดโดยกลุ่มของอนุภาค (Particle Swarm Optimization (PSO))	7
2.2.1 อนุภาค (Particle)	7
2.2.2 อัลกอริทึมสำหรับการจัดระเบียบกลุ่มของอนุภาค	8
2.3 ความรู้ตามหลักสูตรซึ่งถูกนำมาใช้หรือบูรณาการในโครงการ	10
2.4 ความรู้นอกหลักสูตรซึ่งถูกนำมาใช้หรือบูรณาการในโครงการ	10
3 โครงสร้างของโครงการ	11
3.1 การจัดเก็บข้อมูล	11
3.2 การสร้างตัวชี้วัดทางเทคนิคด้วย Fuzzy Logic	12
3.2.1 ตัวแปรทางภาษา (Linguistic Variable)	12
3.2.2 Fuzzy Rules	12
3.3 การปรับแต่ง Fuzzy Logic ด้วย PSO	14
3.3.1 กลยุทธ์ที่เราใช้ปรับแต่ง	14
3.3.2 Backtesting	14
3.3.3 Objective Function	15
3.4 การจัดการเงินทุน	16
3.5 เว็บไซต์ฟเวอร์	16
3.6 การพัฒนาเว็บไซต์และแอปพลิเคชันโทรศัพท์	17
3.7 แผนภาพกระแสข้อมูลโดยรวมของระบบ (Data Flow Diagram)	17
บรรณานุกรม	18
ก ความคืบหน้าในการพัฒนา	20

สารบัญรูป

2.1	พีชชีเซต หนาว,อบอุ่น,ร้อน และฟังก์ชันภาวะสมาชิก	4
2.2	ตัวอย่างการทำงานของระบบประมวลผลพีชชีลอจิก	5
2.3	ตัวอย่างกราฟฟังก์ชันภาวะสมาชิก Triangular function	5
2.4	ตัวอย่างการอนุมาน	7
2.5	รูปแบบของเพื่อนบ้านสำหรับการจัดระเบียบกลุ่มของอนุภาค ทอพอโลยีแบบดาว	8
2.6	รูปแบบของเพื่อนบ้านสำหรับการจัดระเบียบกลุ่มของอนุภาค ทอพอโลยีแบบวงแหวน	8
3.1	โครงสร้างของการจัดเก็บข้อมูล โดนเส้นประคือทำครั้งเดียวในตอนแรกเริ่ม และเส้นทึบจะ ทำในทุกๆ ชม. โดยเป็นการเรียกใช้โปรแกรม dBUpdater ใน AWS Lambda	12
3.2	ตัวแปรทางภาษาสำหรับ RSI, Bollinger Band, long, short	13
3.3	ตัวแปรทางภาษาและตัวแปรที่เราต้องการจะปรับแต่ง $\mu_{\text{medium}} = b(1 - \frac{ x-a }{s})$ (ในที่นี้คือ เราจะปรับแต่งค่าของ a, b, s)	14
3.4	ตัวอย่างของ Net Profit และ Maximum Drawdown	15
3.5	แผนภาพกระแสข้อมูล	17
ก.1	ตารางการทำงาน	20
ก.2	ตัวอย่างข้อมูลตลาดหุ้นในฐานข้อมูล	21
ก.3	UI/UX ของแอปพลิเคชันบนโทรศัพท์	22
ก.4	UI/UX ของเว็บไซต์	23

สารบัญตาราง

3.1 ตัวอย่างของ Fuzzy Rules ที่ใช้แค่ RSI และ Bollinger Band เพื่อสร้าง long และ short. 13

บทที่ 1

บทนำ

1.1 ที่มาของโครงการ

ในปัจจุบัน, นักลงทุนมีการใช้การวิเคราะห์ทางเทคนิค (Technical Analysis) เพื่อช่วยให้การซื้อขายสินทรัพย์ในระยะสั้นได้กำไรสูงสุดเท่าที่เป็นไปได้ ซึ่งก็มักจะมีการใช้ตัวชี้วัดทางเทคนิค (Technical Indicators) หลายๆ อัน ในการที่จะพยายามหาจุดเข้าซื้อ หรือจุดขาย โดย ตัวชี้วัดทางเทคนิคเหล่านี้ส่วนใหญ่แล้วเป็นการคำนวณทางสถิติที่ใช้ ราคาย้อนหลัง, ปริมาณการซื้อขายย้อนหลัง, หรืออื่นๆ ในการ คำนวณค่ามาเพื่อที่จะพยายามทำนายทิศทางของตลาด ซึ่งเราสามารถตีความหมายค่าของตัวชี้วัดทางเทคนิคด้วยเกณฑ์บางอย่าง เช่น สำหรับ RSI (Relative Strength Index) วิธีตีความหมายโดยทั่วไปคือ ถ้า RSI มากกว่า 70 หมายความว่าตลาดอยู่ในภาวะซื้อมากเกินไปให้ขาย และถ้า RSI น้อยกว่า 30 หมายความว่าตลาดอยู่ในภาวะขายมากเกินไปให้เข้าซื้อ

ผู้จัดทำคิดว่าสามารถทำได้ดีกว่าการตีความหมายแบบในตัวอย่างก่อนหน้านี้ โดยใช้ Fuzzy Rule ในการตีความหมายจะให้ผลลัพธ์ที่ดีกว่าเนื่องจากตลาด ซื้อขายสินทรัพย์นั้นมีความผันผวนและไม่แน่นอน ซึ่ง Fuzzy Logic นั้นสามารถทำงานได้ดีในการตีความ และใช้ข้อมูลที่คลุมเครือและไม่แน่นอน นอกจากนี้ในงานวิจัยของ [1] ก็มีการใช้ Fuzzy Logic ในการระบบการซื้อขายสินทรัพย์สำหรับจังหวะการเข้าซื้อ และการจัดการเงินทุน ซึ่งทำงานได้ดีในตลาด NASDAQ100 และ EUROSTOXX ใน [2] ก็มีการใช้ Fuzzy Logic ในการสร้างตัวชี้วัดทางเทคนิคจากการรับความเสี่ยงของผู้ใช้, ข้อมูลของตลาด, และอื่นๆ ซึ่งได้ผลลัพธ์ว่าตัวชี้วัดทางเทคนิคจาก Fuzzy Logic มีประสิทธิภาพมากกว่าตัวชี้วัดทางเทคนิคแบบปกติ ได้แก่ MA, RSI และ MACD

ผู้จัดทำจึงได้สร้างระบบในการสร้างตัวชี้วัดทางเทคนิคใหม่จากตัวชี้วัดทางเทคนิค เช่น MACD, RSI, และอื่นๆ ด้วย Fuzzy Logic และสร้างระบบการจัดการเงินทุนด้วย optimal-F ที่ดัดแปลงให้ใช้ตัวชี้วัดทางเทคนิคที่มาจาก Fuzzy Logic (อ้างอิงจาก [1]) เพื่อช่วยในการตัดสินใจซื้อขายสินทรัพย์ให้ได้กำไรมากยิ่งขึ้น โดยระบบทั้งหมดนี้จะมีเว็บไซต์ และแอปโทรศัพท์แอนดรอยด์เป็นอินเตอร์เฟซในการใช้งาน โดยผู้จัดทำจะทำตัวชี้วัดจาก Fuzzy Logic นั้นบน 2 ตลาดก็คือตลาดหุ้น NASDAQ และตลาด Crypto-Currency เพื่อเปรียบเทียบความแตกต่างของผลลัพธ์ในตลาดที่มีความผันผวนต่างกัน และมีความถี่ของข้อมูลที่แตกต่างกัน

1.2 วัตถุประสงค์ของโครงการ

1. เพื่อพัฒนา Fuzzy Logic ร่วมกับ Particle Swarm Optimization (PSO) สำหรับการสร้างวิธีการซื้อขายเฉพาะของแต่ละสินทรัพย์
2. เพื่อสร้างเว็บไซต์และแอปโทรศัพท์เพื่อให้ผู้ใช้งานสามารถใช้งานระบบได้

1.3 ขอบเขตของโครงการ

ข้อมูลที่ใช้งานคือข้อมูลของตลาดหุ้น NASDAQ ที่ได้จาก AlphaVantage (และ Finnhub) ในช่วงประมาณไตรมาสแรกของปี 2021 ถึงปัจจุบัน โดยมีของบริษัท TSLA, NKE, และ JPM และข้อมูลของตลาด Crypto-Currency จาก Binance โดยมี BTC, ETH และ BNB ในช่วงตั้งแต่ที่ Binance มีข้อมูลให้ รูป

แบบของข้อมูลจะอยู่ในรูปของแท่งเทียนซึ่งมี ราคาเปิด, ราคาสูงสุด, ราคาต่ำสุด, ราคาปิด, และปริมาณการซื้อขาย ในช่วงเวลา 1 ชั่วโมง

1.4 ประโยชน์ที่ได้รับ

เว็บไซต์และแอปโทรศัพท์ที่สามารถใช้ตัวชี้วัดทางเทคนิคจาก Fuzzy Logic ของเราทั้งที่ได้มาจากการปรับแต่งด้วย PSO และแบบที่จัดทำขึ้นมาเอง โดยมี UI ให้ user ปรับแต่ง Fuzzy Logic ต่างๆ เองได้ และมีราคาสินทรัพย์ที่อยู่ในรูปแบบแท่งเทียนโชว์อยู่ด้วย

1.5 เทคโนโลยีและเครื่องมือที่ใช้

1. Rocket (Web Server Framework), Rust: สำหรับพัฒนาในส่วนของ Backend, การฝึกสอน Model, และ API ไว้ติดต่อกับ Frontend
2. SvelteKit (Web Application Framework), Typescript: สำหรับพัฒนา Frontend ในส่วนของหน้าเว็บไซต์
3. Flutter (Mobile Application Framework): สำหรับพัฒนา Frontend ในส่วนของเว็บโทรศัพท์
4. MongoDB: สำหรับเก็บข้อมูลตลาดสินทรัพย์ที่เอาไว้ใช้ในการฝึกสอน Model, ใช้ในการแสดงบน Frontend และเก็บ Model ที่ฝึกสอนแล้ว

1.6 แผนการดำเนินงาน

Operation Steps	JAN 2023				FEB 2023				MAR 2023				Progress
	W1	W2	W3	W4	W1	W2	W3	W4	W1	W2	W3	W4	
Experiment with backend code.													70%
Experiment with frontend code.													50%
Design UI/UX													80%
Prototyping													50%
Summarized as a report													90%

1.7 บทบาทและความรับผิดชอบ

- นายธนนต์ ตั้งอัน รหัส 630610737 ทำในส่วนของ Backend โดยมีองค์ประกอบหลักๆ ก็คือตัวเว็บเซิร์ฟเวอร์, database, การคำนวณ fuzzy logic และตัวชี้วัดทางเทคนิคต่างๆ และการปรับแต่ง fuzzy logic ด้วย PSO

- นายธนวัฒน์ บำเพ็ญพันธุ์ รหัส 630610736 ทำในส่วนของ Frontend คือการออกแบบ UI/UX, สร้างเว็บและแอปพลิเคชันมือถือเพื่อติดต่อกับ User และบริการเว็บเซิร์ฟเวอร์

1.8 ผลกระทบด้านสังคม สุขภาพ ความปลอดภัย กฎหมาย และวัฒนธรรม

ระบบนี้อาจจะสามารถต่อเติมด้วยการใส่ตัวชี้วัดทางเทคนิคอื่นๆ ที่อาจจะมาจากแหล่งต่างๆมาเพิ่มความละเอียดในการวิเคราะห์บางอย่าง ซึ่งถ้าระบบนี้สำเร็จ ระบบนี้อาจจะเป็นเครื่องมือสำคัญให้กับนักลงทุนหลายๆคน และสามารถช่วยสร้างกำไรให้นักลงทุนเพิ่มได้

บทที่ 2

ทฤษฎีที่เกี่ยวข้อง

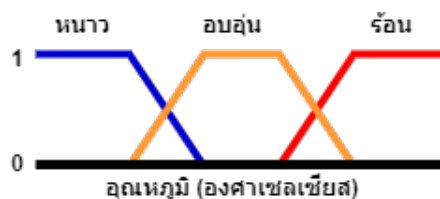
การทำโครงงาน เริ่มต้นด้วยการศึกษาค้นคว้า ทฤษฎีที่เกี่ยวข้อง หรือ งานวิจัย/โครงงาน ที่เคยมีผู้เสนอไว้แล้ว ซึ่งเนื้อหาในบทนี้ก็จะเกี่ยวกับการอธิบายถึงสิ่งที่เกี่ยวข้องกับโครงงาน เพื่อให้ผู้อ่านเข้าใจเนื้อหาในบทถัดๆ ไปได้ง่ายขึ้น

2.1 ฟัชซีลอจิก (Fuzzy Logic)

ฟัชซีลอจิก เป็นแนวคิดเกี่ยวกับการวิเคราะห์เชิงตรรกะ แต่การวิเคราะห์ไม่ได้มีเพียง ถูกกับผิด หรือ 0 กับ 1 เนื่องจากเหตุการณ์ในความเป็นจริงสร้างความคลุมเครือในการวิเคราะห์ เช่น อุณหภูมิอากาศ 20 องศาเซลเซียสเป็นอากาศที่หนาวไปหรือไม่? หากนำคำถามนี้ไปให้ผู้วิเคราะห์ต่างที่อยู่อาศัยกัน จะได้คำตอบที่ไม่เหมือนกัน เนื่องจากการวิเคราะห์แบบนี้ไม่เหมาะกับการตอบเพียงใช่หรือไม่ การใช้ฟัชซีลอจิก (Fuzzy Logic) มาใช้วิเคราะห์เหตุการณ์จึงจะได้คำตอบที่ดีกว่า แทนที่จะตอบเพียงแค่ ใช่หรือไม่ คำตอบที่ได้จะเป็นพจน์ของ ตัวแปรทางภาษา (Linguistic Variable) และความเป็นสมาชิก เช่นตัวแปรทางภาษาอุณหภูมิมีค่า หนาว 60% อุ่น 15% ร้อน 0% (เพราะผู้วิเคราะห์อาจจะรู้สึกหนาวแต่ก็ไม่ได้หนาวเกินไปหรืออบอุ่นอยู่เล็กน้อย) จะเห็นว่าการบอกค่าเชิงตรรกะแบบฟัชซีสะท้อนความจริงได้ดีกว่าการตอบแบบเดิม

2.1.1 ฟัชซีเซต (Fuzzy Set)

เป็นเซตที่ขอบเขตไม่เด่นชัดหรือคลุมเครือโดยการบอกค่าเชิงตรรกะจะถูกสร้างเป็นฟัชซีเซตที่เราสามารถวัดระดับความเป็นสมาชิก (Membership Value) ของสมาชิกในเอกภพสัมพัทธ์ต่อฟัชซีเซตนั้นผ่านทางฟังก์ชันภาวะสมาชิก (Membership function) ซึ่งเป็นฟังก์ชันที่รับสมาชิกในเอกภพสัมพัทธ์แล้วส่งไปที่ช่วง $[0,1]$ โดยจากตัวอย่างดังกล่าวจะสามารถสร้างเป็นฟัชซีเซตได้เป็น เซตของอากาศ หนาว, อบอุ่น, ร้อน โดยให้อุณหภูมิเป็นสมาชิกของเซตซึ่งสมาชิกแต่ละตัวสามารถเป็นสมาชิกของทุกเซตได้ เช่น อุณหภูมิ 20 องศาเซลเซียส มีระดับความเป็นสมาชิกในฟัชซีเซตอากาศหนาว 0.6, อุ่น 0.15, ร้อน 0

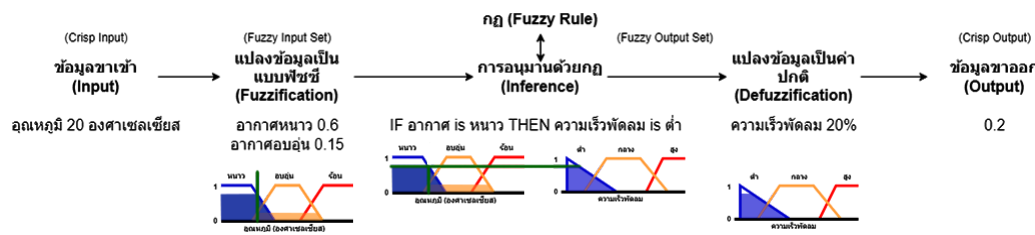


รูปที่ 2.1: ฟัชซีเซต หนาว,อบอุ่น,ร้อน และฟังก์ชันภาวะสมาชิก

2.1.2 ระบบประมวลผลฟัชซีลอจิก (Fuzzy Logic System)

เป็นการนำเอาความสามารถของฟัชซีลอจิกมาสร้างเป็นระบบประมวลผลแบบฟัชซีลอจิกซึ่งเป็นการเลียนแบบการคิด การหาเหตุผล การตัดสินใจและการกระทำของมนุษย์ โดยจะมีส่วนประกอบสำคัญ 4 ส่วนคือ 1. การแปลงข้อมูลเข้าเป็นฟัชซี (Fuzzification), 2. กฎ (Fuzzy Rules), 3. การอนุมานหรือการตี-

ความ (Inference), 4. การแปลงข้อมูลฟัซซีเป็นตัวเลข (Defuzzification) ซึ่งจะมีตัวอย่างการทำงานเมื่อใช้ระบบประมวลผลฟัซซีลอจิกดังภาพรวมในรูปที่ 2.2



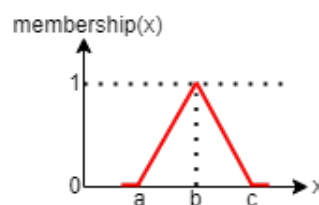
รูปที่ 2.2: ตัวอย่างการทำงานของระบบประมวลผลฟัซซีลอจิก

โดยในงานนี้เราใช้ระบบฟัซซีแบบ Mamdani

การแปลงข้อมูลขาเข้าเป็นฟัซซี (Fuzzification)

เป็นการแปลงข้อมูลอินพุตทั่วไปที่เป็นตัวเลข (Crisp Set) ไปเป็นข้อมูลในรูปแบบฟัซซีเซต หรือที่เรียกว่าตัวแปรทางภาษา (Linguistic Variable) โดยจะสร้างฟังก์ชันภาวะสมาชิกซึ่งขึ้นอยู่กับลักษณะของข้อมูลขาเข้าและความสำคัญต่อข้อมูลเอาต์พุต

$$\mu_A(x) = \begin{cases} 0 & \text{if } x \leq a \\ \frac{x-a}{b-a} & \text{if } a < x < b \\ \frac{c-x}{c-b} & \text{if } b \leq x < c \\ 0 & \text{if } x \geq c \end{cases} \quad (2.1)$$



รูปที่ 2.3: ตัวอย่างกราฟฟังก์ชันภาวะสมาชิก Triangular function

กฎฟัซซี (Fuzzy Rules) [3]

เป็นส่วนของการกำหนดวิธีการควบคุมซึ่งได้มาจากผู้เชี่ยวชาญหรือการปรับแต่งทดลองขึ้นเองโดยจะอยู่ในรูปแบบของชุดข้อมูลแบบกฎของภาษา ซึ่งกฎฟัซซีแบบที่นิยมใช้มากและใช้ในงานนี้เป็นกฎฟัซซีแบบ ถ้า-แล้ว (If-then rule) โดยในงานนี้ได้ใช้วิธีการของ Mamdani หากมีอินพุต X_1, X_2, \dots, X_n และพจน์ภาษา $T(x_i)$ ของตัวแปรทางภาษา x_i ในเซตสากล X_i สำหรับ $1 \leq i \leq n$ ในขณะเดียวกัน Y ก็ถูกนิยามด้วยตัวแปรทาง

ภาษา และพจน์ภาษา $T(y)$ ของตัวแปรทางภาษา y ในเซตสากล Y

$$IF x_1 \text{ is } A^{(1)} \text{ and } x_2 \text{ is } A^{(2)} \text{ and } \dots \text{ and } x_n \text{ is } A^{(n)} \text{ THEN } y \text{ is } B$$

โดยที่ $A^{(1)}, A^{(2)}, \dots, A^{(n)}$ เป็นพจน์ในภาษา $T(x_i)$ และ B เป็นพจน์ในภาษา $T(y)$

การอนุมานหรือการตีความ (Inference) [3]

เป็นส่วนของการประมวลผลจะมีการตีความตามเงื่อนไขที่กำหนดไว้ หรือก็คือตีความผ่านกฎฟัซซี ซึ่งจากกฎฟัซซีดังกล่าวจะประกอบด้วยสองส่วนคือ ส่วนที่เกิดขึ้นก่อน (If Part) และผลที่ตามมา (Then part) โดยที่อินพุตและเอาต์พุตนั้นอาจมีหลายตัวก็ได้ขึ้นอยู่กับวิธีการออกแบบ ผลที่ตามมาของแต่ละกฎจะถูกรวมกันด้วยวิธีการตรรกศาสตร์เพื่อให้ได้ค่าเอาต์พุตเพียงค่าเดียว

โดยจะเริ่มจากการหาระดับความเข้ากันได้ของแต่ละอินพุต ($x_i, i \in \{1, 2, \dots, n\}$) กับพจน์ภาษาในกฎนั้น และเนื่องจากลักษณะของส่วนที่เกิดขึ้นก่อน (If Part) ของกฎต้องการให้ทุกอินพุตเป็นไปตามพจน์ภาษาดังนั้นค่าความเป็นสมาชิกของแต่ละอินพุตในแต่ละพจน์ภาษาจะถูกรวมกันในลักษณะของตัวเชื่อม conjunction นั่นคือที่กฎ j

$$\alpha_j = \min\{A_{i1,j}^{(1)}(x_1), A_{i2,j}^{(2)}(x_2), \dots, A_{in,j}^{(n)}(x_n)\} \quad (2.2)$$

และเอาต์พุตของกฎ j เป็นฟัซซีเซตที่เกิดขึ้นจากการตัด (cut off) พจน์ภาษา $B_{i,j}$ ด้วย α_j หรือ

$$OUT_{x_1, x_2, \dots, x_n}^{(j)}(y) = \min(A_{i1,j}^{(1)}(x_1), A_{i2,j}^{(2)}(x_2), \dots, A_{in,j}^{(n)}(x_n), B_{i,j}(y)) \quad (2.3)$$

และเมื่อได้เอาต์พุตของแต่ละกฎแล้ว ฟัซซีเอาต์พุตจากทุกกฎจะถูกรวมกันโดยการหาฟัซซียูเนียนมาตรฐาน (ซึ่งจะได้ฟัซซีเอาต์พุตรวม (OUT)) สมมติให้มีกฎทั้งหมด k กฎ จะได้ OUT เป็น

$$OUT_{x_1, x_2, \dots, x_n}(y) = \max_{j \in \{1, 2, \dots, k\}} \min(A_{i1,j}^{(1)}(x_1), A_{i2,j}^{(2)}(x_2), \dots, A_{in,j}^{(n)}(x_n), B_{i,j}(y)) \quad (2.4)$$

ตัวอย่าง สมมติให้ระบบมีกฎ 2 กฎ โดยที่แต่ละกฎจะมีอินพุต 2 อินพุต และแต่ละอินพุตในแต่ละกฎมีพจน์ภาษาดังรูป โดยที่มีกฎดังนี้คือ

$$R1 : IF x_1 \text{ is } L_1 \text{ and } x_2 \text{ is } H_2, \text{ THEN } y \text{ is } L$$

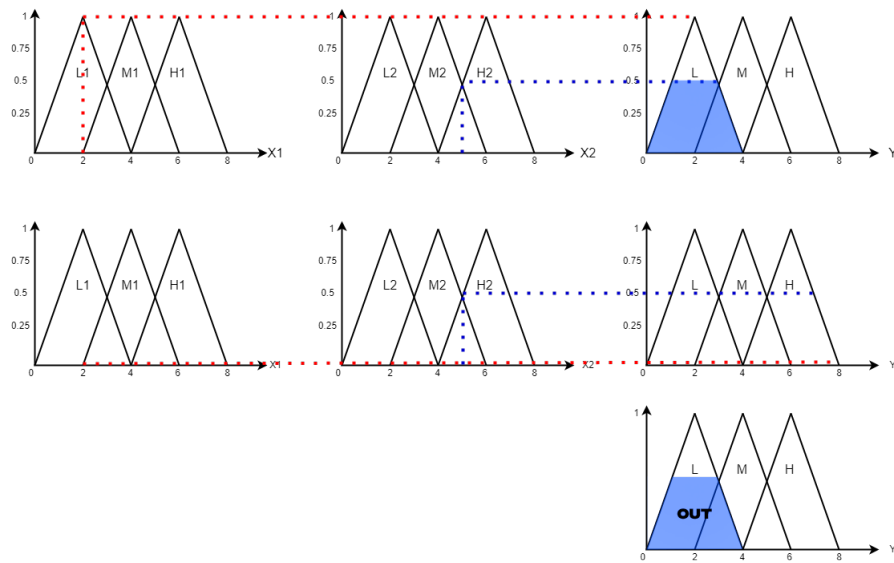
$$R2 : IF x_1 \text{ is } M_1 \text{ and } x_2 \text{ is } M_2, \text{ THEN } y \text{ is } H$$

ถ้ากำหนดให้ x_1 มีค่าเท่ากับ 2 และ x_2 มีค่าเท่ากับ 5 จะได้ว่า

$$\alpha_1 = \min(L_1(x_1), H_2(x_2)) = \min(1, 0.5) = 0.5$$

$$\alpha_2 = \min(M_1(x_1), H_2(x_2)) = \min(0, 0.5) = 0$$

ฟังก์ชันเอาต์พุตของกฎที่ 1 และ 2 และฟังก์ชันเอาต์พุตรวม (OUT) ดังที่แสดงในรูป 2.4



รูปที่ 2.4: ตัวอย่างการอนุมาน

การแปลงข้อมูลฟัซซีเป็นค่าปกติ (Defuzzification)

เนื่องจากผลลัพธ์ที่ได้จากการตีความนั้นยังอยู่ในรูปแบบของฟัซซี ในส่วนนี้เป็นการทำการแปลงข้อมูลที่อยู่ในรูปแบบฟัซซีเป็นข้อมูลที่เป็นตัวเลข (Crisp set) ด้วยวิธีทางคณิตศาสตร์ เช่น Center of Area (Centroid) เพื่อนำค่าที่ได้มาใช้ในการตัดสินใจและนำไปควบคุมระบบได้

วิธีแปลงโดยการหา Centroid จะหาค่าเอาต์พุตจากจุดศูนย์กลางของพื้นที่กราฟที่ได้ตั้งสมการต่อไปนี้

$$de_y = \frac{\int B(z) \cdot z dz}{\int B(z) dz} \quad (2.5)$$

โดย $B(z)$ คือ ค่าความเป็นสมาชิก (Membership Value) ของตำแหน่ง z

2.2 การหาค่าที่เหมาะสมที่สุดโดยกลุ่มของอนุภาค (Particle Swarm Optimization (PSO))

จาก [3] การหาค่าที่เหมาะสมที่สุดโดยกลุ่มของอนุภาค เป็นอัลกอริทึมการค้นหาที่ขึ้นกับประชากร ซึ่งเป็นการจำลองพฤติกรรมเชิงสังคมของฝูงนก ทำทางของฝูงนกเชิงภูมิศาสตร์ที่คาดเดาไม่ได้ โดยที่มีจุดประสงค์ในการค้นพบรูปแบบที่ควบคุมความสามารถของนกในการบินพร้อมกันและสามารถเปลี่ยนทิศทางได้อย่างกะทันหัน โดยการรวมกลุ่มกันใหม่ในลักษณะที่เหมาะสมที่สุด ทำให้เกิดอัลกอริทึมสำหรับการจัดระเบียบกลุ่มของอนุภาค ที่ง่ายและมีประสิทธิภาพ

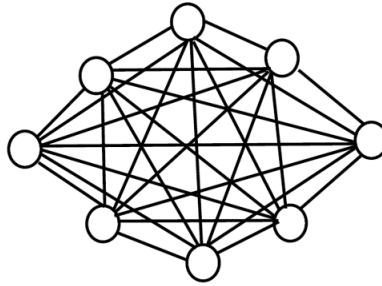
2.2.1 อนุภาค (Particle)

อนุภาค 1 อนุภาค คือคำตอบที่เป็นไปได้ของปัญหาการหาค่าที่เหมาะสม โดยอนุภาคจะบินในปริภูมิการค้นหาหลายมิติ การเปลี่ยนแปลงของอนุภาคในกลุ่มนั้นมีอิทธิพลมาจากประสบการณ์ หรือความรู้ของเพื่อนบ้าน รูป

ร่างของเพื่อนบ้านมีหลายรูปแบบ และมีการสร้างอัลกอริทึมตามแต่ละรูปแบบ

ทอพอโลยีแบบดาว (Star Topology)

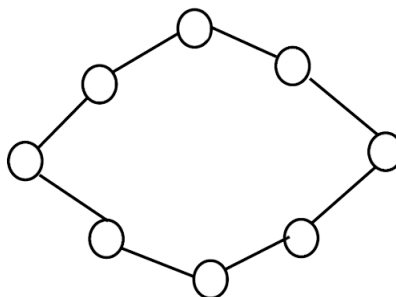
รูปแบบนี้ทำให้แต่ละอนุภาคสามารถติดต่อกับอนุภาคอื่นได้ทุกอนุภาค แต่แต่ละอนุภาคจะสนใจอนุภาคที่ดีที่สุดในกลุ่ม และแต่ละอนุภาคจะเลียนแบบอนุภาคที่ดีที่สุดในกลุ่มนี้เอง โดยอัลกอริทึมที่จำลองสถานการณ์นี้คือ อัลกอริทึมที่ดีที่สุดแบบรวม (global best)



รูปที่ 2.5: รูปแบบของเพื่อนบ้านสำหรับการจัดระเบียบกลุ่มของอนุภาค ทอพอโลยีแบบดาว

ทอพอโลยีแบบวงแหวน (Ring Topology)

รูปแบบนี้ทำให้แต่ละอนุภาคจะติดต่อกับเพื่อนบ้านที่ใกล้ที่สุด n อนุภาค ดังแสดงในรูป เมื่อ $n = 2$ ดังนั้นอนุภาคเคลื่อนที่ตามเพื่อนที่ดีที่สุดในกลุ่มเพื่อนบ้านที่ติดต่อกับได้ ซึ่งอัลกอริทึมที่จำลองสถานการณ์นี้คือ อัลกอริทึมที่ดีที่สุดแบบเฉพาะที่ (local best)



รูปที่ 2.6: รูปแบบของเพื่อนบ้านสำหรับการจัดระเบียบกลุ่มของอนุภาค ทอพอโลยีแบบวงแหวน

2.2.2 อัลกอริทึมสำหรับการจัดระเบียบกลุ่มของอนุภาค

อนุภาคจะบินอยู่ในปริภูมิการค้นหาลายมิติ โดยที่ตำแหน่งของอนุภาคจะเปลี่ยนไปตามประสบการณ์ของตัวเอง อนุภาคเอง หรือของเพื่อนบ้าน ให้ $x_i(t)$ เป็นตำแหน่งของอนุภาค P_i ในปริภูมิไฮเปอร์ (hyperspace) ที่เวลา

t และตำแหน่งของอนุภาคจะเปลี่ยนได้โดยการเพิ่มความเร็ว $v_i(t)$ ให้กับตำแหน่งปัจจุบันดังนี้

$$x_i(t) = x_i(t-1) + v_i(t) \quad (2.6)$$

ซึ่งความเร็วนี้เป็นตัวขับเคลื่อนกระบวนการหาค่าที่เหมาะสม และสะท้อนถึงการแลกเปลี่ยนข้อมูลในสังคม

ฟังก์ชันจุดประสงค์ (Objective Function)

เป็นฟังก์ชันที่เราสร้างขึ้นหรือฟังก์ชันปัญหาที่เราต้องการหาค่าที่เหมาะสมที่สุดเพื่อที่จะให้ได้คำตอบที่ดีที่สุด และจะใช้คำนวณหาความเหมาะสม ซึ่งเปรียบเหมือนประสิทธิภาพของแต่ละอนุภาค

อัลกอริทึมที่ดีที่สุดแบบรวม (Global Best)

อัลกอริทึม gbest นี้เป็นการใช้โครงสร้างทอพอโลยีแบบดาว ดังนั้นการเคลื่อนที่ของอนุภาคจะขึ้นอยู่กับตำแหน่งที่ดีที่สุดของอนุภาคตัวที่ดีที่สุดในกลุ่ม และประวัติจากประสบการณ์ของตนเอง ดังนั้นอัลกอริทึมนี้สามารถสรุปได้ดังนี้

1. ตั้งค่ากลุ่ม ($P(t)$ ที่ $t=0$) ของอนุภาค โดยที่ตำแหน่ง ($x_i(t)$) ของอนุภาค $i (P_i \in P(t))$ จะถูกสุ่มโดยให้ค่าอยู่ภายในปริภูมิไฮเปอร์ ที่ต้องการค้นหาคำตอบ
2. คำนวณค่าประสิทธิภาพ F ของแต่ละอนุภาค โดยใช้ตำแหน่งปัจจุบัน $x_i(t)$
3. เปรียบเทียบค่าที่ได้ในข้อ 2 ของอนุภาค i กับค่าที่ดีที่สุดของตนเอง ($pbest_i$) ดังนี้ ถ้า $F(x_i(t)) < pbest_i$ แล้วกำหนดให้ $pbest_i = F(x_i(t))$ และ $x_{pbest_i}(t) = x_i(t)$
4. เปรียบเทียบค่าที่ได้ในข้อ 2 ของอนุภาค i กับค่าที่ดีที่สุดของกลุ่ม ($gbest$) ดังนี้ ถ้า $F(x_i(t)) < gbest$ แล้วกำหนดให้ $gbest = F(x_i(t))$ และ $x_{gbest}(t) = x_i(t)$
5. ปรับความเร็วของแต่ละอนุภาคดังนี้

$$v_i(t) = v_i(t-1) + \rho_1(x_{pbest_i} - x_i(t)) + \rho_2(x_{gbest} - x_i(t)) \quad (2.7)$$

โดยที่ ρ_1 และ ρ_2 เป็นค่าที่ถูกสุ่มมา

6. ปรับตำแหน่งของแต่ละอนุภาค ตามสมการที่ 2.6 และตั้งค่า $t = t + 1$
7. กลับไปยังข้อ 2 และทำซ้ำ จนกระทั่งจะลู่เข้า (converge)

อัลกอริทึมที่ดีที่สุดแบบเฉพาะที่ (Local Best)

อัลกอริทึม lbest นี้เป็นการใช้เพื่อนบ้านในลักษณะของทอพอโลยีแบบวงแหวน ดังนั้นอนุภาคที่มีผลต่อการเคลื่อนที่คืออนุภาคที่อยู่ในเพื่อนบ้านที่ดีที่สุดและตำแหน่งที่ดีที่สุดของตนเอง ซึ่งอัลกอริทึมนี้จะคล้ายกับแบบ gbest เพียงแต่ในขั้นตอนที่ 4 และ 5 เปลี่ยนจาก gbest เป็น lbest นั่นเอง

อัลกอริทึม lbest นี้จะช้าในการลู่เข้ามากกว่าแบบ gbest แต่จะให้คำตอบที่ดีกว่า และเป็นการค้นหาโดยครอบคลุมพื้นที่ได้กว้างกว่า

2.3 ความรู้ตามหลักสูตรซึ่งถูกนำมาใช้หรือบูรณาการในโครงการ

ทฤษฎีฟuzzyลอจิก และการหาคำตอบที่เหมาะสมแบบฟuzzyอนภาค ทั้ง 2 ทฤษฎีนี้เป็นสิ่งที่เราได้เรียนรู้มาจากวิชา Introduction to Computational Intelligence for Computer Engineering (261456) โดยในงานนี้เราได้นำทั้ง 2 ทฤษฎีมาใช้งานร่วมกันโดยใช้ทฤษฎีการหาคำตอบที่เหมาะสมแบบฟuzzyอนภาคในการปรับพารามิเตอร์ในระบบประมวลผลฟuzzy

2.4 ความรู้นอกหลักสูตรซึ่งถูกนำมาใช้หรือบูรณาการในโครงการ

ความรู้เกี่ยวกับการเทรต การใช้งานและวิเคราะห์ตัวชี้วัดทางเทคนิค

บทที่ 3

โครงสร้างของโครงการ

ในบทนี้จะกล่าวถึงหลักการ, การนำทฤษฎีที่เกี่ยวข้องมาประยุกต์ใช้ และการออกแบบของระบบ

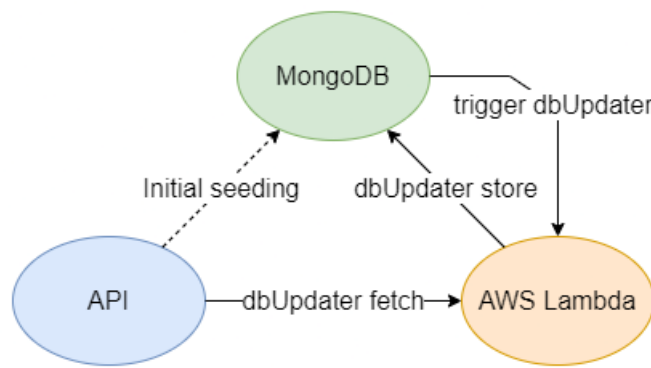
3.1 การจัดเก็บข้อมูล

โดยข้อมูลราคาหุ้นทุกตัวจะมีแหล่งที่มาจาก 2 ที่ก็คือ AlphaVantage และ Finnhub โดย AlphaVantage จะให้ข้อมูลย้อนหลังไป 2 ปี ส่วน Finnhub จะเอาไว้ใช้อัปเดต ข้อมูลแบบ real-time ทุกๆ 1 ชม. และในส่วนของราคา Crypto-Currency จะมีแหล่งที่มาจาก Binance ทั้งหมด เราใช้ MongoDB เป็น Database สำหรับจัดเก็บข้อมูลตลาดหุ้น และตัวชี้วัดทางเทคนิคที่เราต้องการใช้ เช่น RSI, MA เป็นต้น

ในตอนเริ่มต้นนั้นเราดึงข้อมูลที่ต้องการมาจาก AlphaVantage API ซึ่งได้มาเป็นข้อมูลตลาดหุ้นย้อนหลัง 2 ปีโดย และเก็บข้อมูลลง MongoDB ด้วย Rust โดยมีการแปลงข้อมูลให้เป็นในรูปแบบข้อมูลตลาดของเราซึ่งก็จะประกอบด้วย

1. ticker: ชื่อของหุ้นที่ทำการซื้อขาย เช่น AAPL/USD, TSLA/USD, ETH/USDT
2. open: เป็นราคาซื้อขายแรกที่เกิดขึ้นใน ช่วงเวลานั้นๆ
3. close: เป็นราคาสุดท้ายที่เกิดขึ้นจากการซื้อขายสิ้นสุด ของช่วงเวลานั้นๆ
4. high: การเคลื่อนไหวของราคาหุ้น ณ ระดับราคาสูงสุดในช่วงเวลานั้นๆ
5. low: การเคลื่อนไหวของราคาหุ้น ณ ระดับราคาต่ำสุดในช่วงเวลานั้นๆ
6. volume: ปริมาณการซื้อขายในช่วงเวลานั้นๆ

จากนั้นในการอัปเดตข้อมูลแบบ real-time เราจะใช้ MongoDB Scheduled Triggers ที่จะไปเรียกใช้ AWS Lambda ที่เราสร้างขึ้นมาโดยใน Lambda จะดึงข้อมูลจาก Finnhub มาอัปเดตที่จะมีการอัปเดตข้อมูลของตัวชี้วัดทางเทคนิคที่เราต้องการใช้ด้วย และอัปเดตตัวชี้วัดทางเทคนิคที่สร้างจาก fuzzy logic ของเราเองด้วย ในส่วนของ Crypto-Currency ก็จะใช้ระบบแบบเดียวกันแต่จะใช้ Binance API ทั้งในการดึงข้อมูลครั้งแรกและการอัปเดตแบบ real-time



รูปที่ 3.1: โครงสร้างของการจัดเก็บข้อมูล โดเนลเส้นประคือทำครั้งเดียวในตอนแรกเริ่ม และเส้นทึบจะทำได้ในทุกๆ ชม. โดยเป็นการเรียกใช้โปรแกรม dbUpdater ใน AWS Lambda

3.2 การสร้างตัวชี้วัดทางเทคนิคด้วย Fuzzy Logic

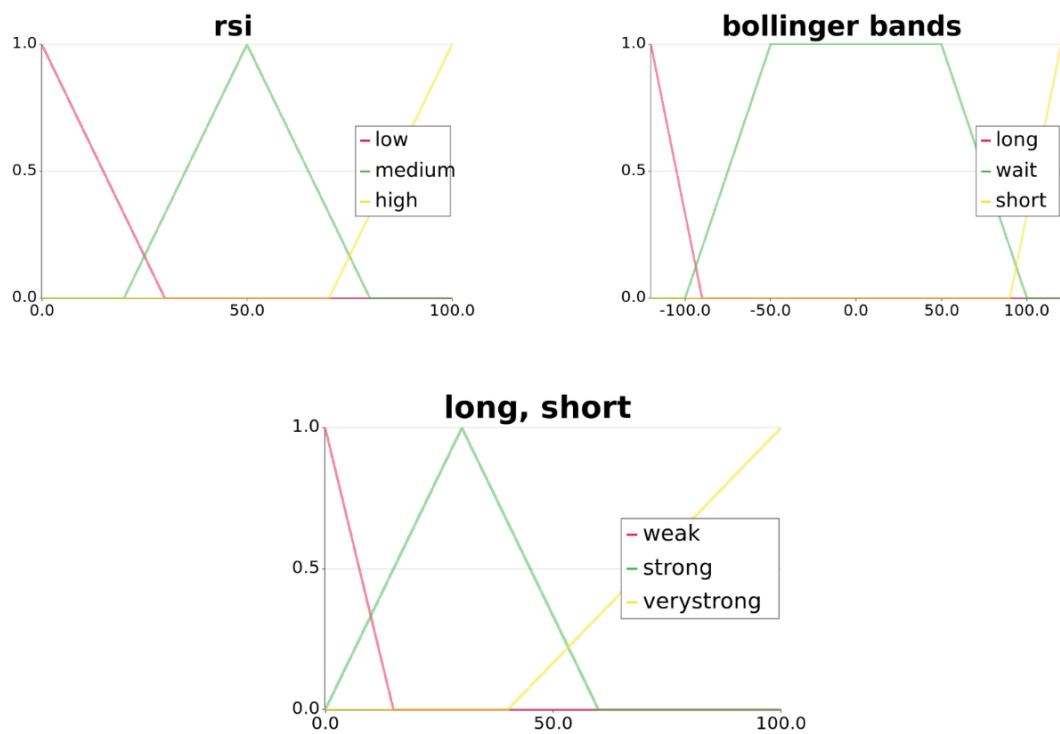
เราจะใช้ Mamdani Fuzzy Inference System กับตัวแปรทางภาษาและ fuzzy rule ที่จะกล่าวด้านล่างนี้ในการคำนวณค่าสัญญาณของเรามา โดย defuzzification method จะใช้แบบ centroid

3.2.1 ตัวแปรทางภาษา (Linguistic Variable)

สำหรับตัวชี้วัดทางเทคนิคแต่ละตัวที่เราได้ให้ไว้ได้แก่ Relative Index Strength (RSI), Bollinger Band, Moving Average Convergence/Divergence (MACD), Average Directional Index (ADX), Aroon oscillator, On-Balance Volume (OBV), Stochastic Oscillator, Accumulation/Distribution Indicator (A/D) เราจะมีตัวแปรทางภาษาสำหรับแต่ละตัวชี้วัดที่สอดคล้องกับการตีความต่างๆ ของมัน และเราก็จะมีตัวแปรทางภาษาที่ใช้ในการสร้างตัวชี้วัดทางเทคนิคใหม่ โดยเราจะทำเป็นสัญญาณ long (ควรเข้า position long) และสัญญาณ short (ควรเข้า position short) โดยทั้ง long และ short จะคิดมาจากตัวชี้วัดทางเทคนิคที่กล่าวถึงด้านบน ยกตัวอย่างตัวแปรทางภาษาที่เราอาจจะใช้บนรูปที่ 3.2 โดเนลในระบบผู้ใช้งานก็สามารถปรับแต่งตัวแปรทางภาษาเหล่านี้ได้ตามต้องการของตัวเองผ่าน website และ mobile application

3.2.2 Fuzzy Rules

เราจะใช้การตีความต่างๆ ของแต่ละตัวชี้วัดมาสร้าง Fuzzy Rule เริ่มต้น ยกตัวอย่างเช่นถ้าเราใช้แค่ RSI และ Bollinger Band ในการสร้าง long และ short เราจะมี fuzzy rule เหมือนในตารางที่ 3.1 โดยในระบบของเราจริงๆ เราจะมีตัวแปรทางภาษาที่เรากล่าวในหัวข้อก่อนหน้านี้มาทั้งหมดสร้าง Fuzzy Rule ในการสร้างสัญญาณ long และ short และเรา จะออกแบบระบบให้ผู้ใช้งานสามารถปรับแต่งกฎตรงนี้ได้ในทั้ง website และ mobile application



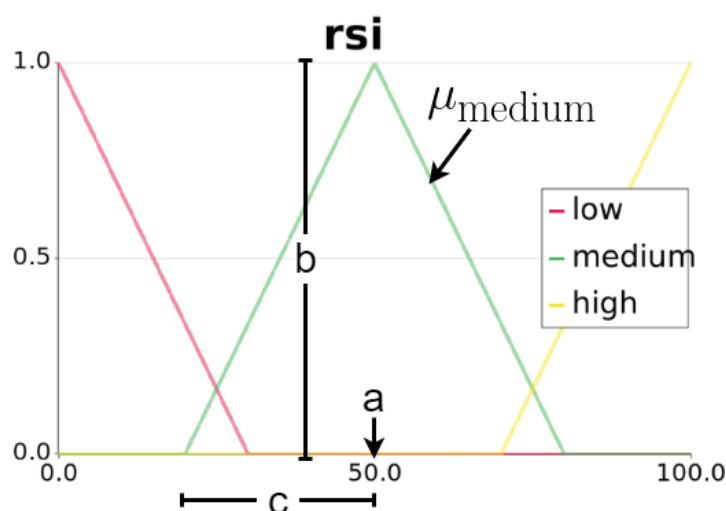
รูปที่ 3.2: ตัวแปรทางภาษาสำหรับ RSI, Bollinger Band, long, short

RSI	Bollinger Bands	LONG	SHORT
HIGH	LONG	WEAK	WEAK
HIGH	WAIT	WEAK	STRONG
HIGH	SHORT	WEAK	VERYSTRONG
MEDIUM	LONG	WEAK	STRONG
MEDIUM	WAIT	WEAK	WEAK
MEDIUM	SHORT	STRONG	WEAK
LOW	LONG	VERYSTRONG	WEAK
LOW	WAIT	STRONG	WEAK
LOW	SHORT	WEAK	WEAK

ตารางที่ 3.1: ตัวอย่างของ Fuzzy Rules ที่ใช้แค่ RSI และ Bollinger Band เพื่อสร้าง long และ short.

3.3 การปรับแต่ง Fuzzy Logic ด้วย PSO

เป้าหมายของเราในการปรับแต่ง Fuzzy Logic ที่ใช้สำหรับการสร้างตัวชี้วัดทางเทคนิคใหม่ของเรา นั้น ก็คือการปรับแต่งตัวแปรทางภาษาต่างๆ ที่มีอยู่ fuzzy rules เพื่อให้ ตัวชี้วัดทางเทคนิคของเรา นั้นสามารถสร้างกำไรได้มากที่สุดในวิธีการเทรดที่เราใช้ปรับแต่ง โดยเราจะใช้ PSO (Particle Swarm Optimization) ในการปรับพารามิเตอร์ที่ใช้ สร้าง linguistic variable แต่ละอัน โดยพารามิเตอร์ในการสร้าง fuzzy set นั้น จะแตกต่างกันไปตามรูปแบบของ fuzzy set เช่นถ้าเป็นแบบสามเหลี่ยมก็จะมีพารามิเตอร์ดังที่เห็นในรูปที่ 3.3 โดยเราจะทำการปรับแต่งนี้และบันทึก fuzzy rules ใหม่ในฐานข้อมูลของเราทุกๆ เดือน



รูปที่ 3.3: ตัวแปรทางภาษาและตัวแปรที่เราต้องการจะปรับแต่ง $\mu_{\text{medium}} = b(1 - \frac{|x-a|}{s})$ (ในที่นี้คือเราจะปรับแต่งค่าของ a, b, s)

3.3.1 กลยุทธ์ที่เราใช้ปรับแต่ง

โดยในการปรับแต่ง Fuzzy Logic ของเรานั้นอันดับแรกเลยเราต้องเลือกกลยุทธ์การเทรดที่เราต้องการปรับแต่ง ให้มีผลต่อตัวชี้วัดทางเทคนิค ยกตัวอย่างกลยุทธ์การเทรด เช่น มีเงินทุน 2000 บาท ถ้า buySignal มากกว่า 50 ให้เข้าซื้อด้วย 100 บาท ด้วย stop-loss ที่ 10% และ take profit ที่ 20%

3.3.2 Backtesting

Backtesting คือการนำกลยุทธ์การเทรดที่เราเลือก ไปใช้กับข้อมูลในอดีตในกรอบเวลาที่ผ่านๆ มาเพื่อทดสอบว่ากลยุทธ์นั้นไปใช้ในตลาดจริงๆ ในอดีตแล้วได้ผลดีแค่ไหน โดยเราสามารถเลือกกรอบเวลาที่ตลาดมีลักษณะคล้ายๆ กับในปัจจุบัน แล้วลองปรับเปลี่ยนและทดสอบกลยุทธ์การเทรดนั้นๆ ได้เพื่อให้ได้ผลลัพธ์ที่เราต้องการ

โดยเราจะทำการ backtest ด้วยกลยุทธ์การเทรดที่เราเลือกมา แล้วเก็บข้อมูลการเทรดที่เกิดขึ้นทั้งหมด โดยแต่ละการเทรดจะมีข้อมูลดังนี้

- เวลาที่เข้า position
- เวลาที่ออก position
- ราคาที่เข้าซื้อ
- ราคาที่ขาย
- จำนวนเงินที่จ่ายไป
- กำไรขาดทุนที่ได้ (realizedPnl)

3.3.3 Objective Function

เราจะใช้ Objective Function ที่คำนวณมาดังนี้ $f = NP - MDD$ โดย

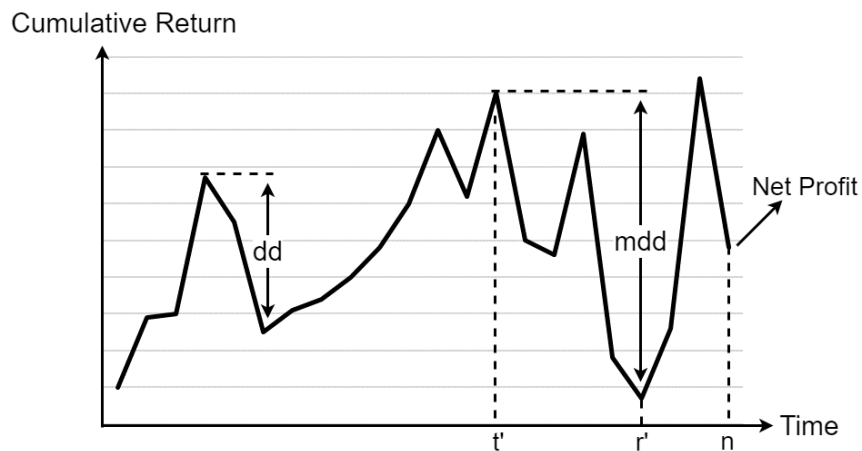
- $NP = \frac{\sum_{i=0}^n p_i(\text{realizedPnl})}{\text{startMoney}}$ คือ Net Profit ที่มีค่าอยู่ในช่วง $[0, \infty)$ ซึ่งได้จากการเทรดทั้งหมด โดยคำนวณจากข้อมูลการเทรดที่เราได้จากการทำ backtest โดย n คือจำนวนข้อมูลทั้งหมด และ $p_i(\text{realizedPnl})$ คือข้อมูลตัวที่ i โดยเอาค่า realizedPnl มา
- MDD (Maximum Drawdown ตัวอย่างในรูปที่ 3.4) มีค่าอยู่ในช่วง $[0, 1]$ โดยเราสามารถคิดค่านี้ได้โดยให้

$$g(x) = \sum_{i=0}^x p_i(\text{realizedPnl})$$

$$MDD' = \max_{r \in (0, n)} \left[\max_{t \in (0, r)} g(t) - g(r) \right] \quad (3.1)$$

แล้วให้เราจำค่า $y = g(t)$ ที่ทำให้ได้ MDD' เยอะที่สุดไว้ แล้วจะได้ว่า $MDD = \frac{MDD'}{y}$

ในส่วนของ hyper parameters ต่างๆ ที่เราต้องตั้งให้ PSO algorithm เช่น จำนวน particles, การคำนวณ velocity เป็นต้น จะเปลี่ยนไปตามแต่ละครั้งของการปรับแต่ง โดยเราจะทดลองหลายๆ แบบเพื่อให้ได้ตัวชี้วัดที่มีประสิทธิภาพดีที่สุด



รูปที่ 3.4: ตัวอย่างของ Net Profit และ Maximum Drawdown

3.4 การจัดการเงินทุน

เราจะใช้ optimal-f ([4]) ที่ถูกดัดแปลงตามที่ [1] ได้ทำไว้ในส่วนของการจัดการเงินทุน ซึ่งจะบอกเราว่าควรลงทุนโดยใช้เงินเท่าไร เพื่อให้เงินกำไรเติบโตแบบ geometric โดยจะคิดมาจากผลลัพธ์ของการเทรดก่อนหน้านี้ ถ้าเราเทรดสำเร็จเยอะก็จะเพิ่มเงินที่จะลงทุน ถ้าเทรดพลาดเยอะก็จะลดเงินที่จะลงทุน

อันดับแรกให้เราหาค่า f ที่ทำให้ terminal wealth relative (TWR) ในสมการ 3.2 มีค่ามากที่สุด

$$\text{TWR}(f) = \prod_{i=1}^n \text{HPR}_i(f) \quad (3.2)$$

$$\text{HPR}_i(f) = 1 + \frac{f \cdot p_i(\text{realizedPnl})}{\text{riskFactor}} \quad (3.3)$$

โดยที่ HPR คือ holding period return หรือก็คืออัตราส่วนกำไรขาดทุนของแต่ละ position, n คือจำนวน position ทั้งหมด, $p_i(\text{realizedPnl})$ คือกำไรขาดทุนของ position ที่ i , และ riskFactor คือค่าสัมบูรณ์ของ $p_i(\text{realizedPnl})$ ที่เยอะที่สุด

แต่ในปรกติแล้วค่า f ที่เราได้มานั้นจะมีความเสี่ยงมากเกินไปเราก็จะใช้เป็น liquid-F ที่เป็น 10% ของ f เป็น $\text{liquid}_f = 0.1f$

$$\text{size} = \text{liquid}_f + \frac{(\text{output} - \text{threshold}) \cdot (f - \text{liquid}_f)}{\text{output}_{\max} - \text{threshold}} \quad (3.4)$$

โดย output คือค่าจากสัญญาณ long หรือ short ของเรา, threshold คือค่าที่ output ที่ต่ำที่สุดที่เราจะเข้า position, และ output_{\max} คือค่าที่มากที่สุดที่เป็นไปได้ของ output จากนั้นเราก็นำ size ไปคำนวณจำนวนที่จะลงทุนด้วยสมการ 3.5

$$\text{amount} = \frac{C \cdot \text{size}}{\text{price}} \quad (3.5)$$

โดย C คือจำนวนเงินที่เราทำไปลงทุนได้ และ price คือราคาของสินทรัพย์ที่เราจะลงทุน แล้วถ้าเรามี C ไม่พอให้เราลงทุนมากที่สุดเท่าที่จะทำได้

3.5 เว็บเซิร์ฟเวอร์

ก่อนจะเรียกใช้ APIs ต่างๆของเรานั้นผู้ใช้งานต้องทำการสร้างบัญชีเอาไว้ก่อนเพื่อให้สามารถเก็บค่าการปรับแต่ง fuzzy logic ที่ผู้ใช้งานแต่ละคนทำได้ แล้ว endpoints แต่ละอันนั้นก็ต้องส่ง token ยืนยันตัวผู้ใช้งานมาด้วย โดยเราจะมี endpoints ดังต่อไปนี้

1. GET /api/ohlcv?symbol=xxx จะให้ข้อมูล OHLC ของสินทรัพย์ที่เราต้องการ
2. GET /api/indicator?symbol=xxx&type=xxx จะให้ข้อมูลของตัวชี้วัดทางเทคนิคที่เราต้องการ เช่น RSI, MACD, และอื่นๆที่กล่าวไป
3. GET /api/tunedfuzzy?symbol=xxx จะให้ข้อมูลของตัวชี้วัดที่สร้างขึ้นมาจาก fuzzy logic และถูกปรับแต่งด้วย PSO แล้ว

4. GET /api/fuzzy?symbol=xxx จะให้ข้อมูลของตัวชี้วัดที่สร้างขึ้นด้วย fuzzy logic ที่ใช้กฎต่างๆ ที่ผู้ใช้ปรับแต่งไว้
5. GET /api/size จะให้ค่า size ในหัวข้อการจัดการเงินทุนมา เพื่อเอาไว้แนะนำขนาดของ position
6. POST /api/fuzzy ใช้ปรับกฎและตัวแปรทางภาษาของ fuzzy logic

3.6 การพัฒนาเว็บไซต์และแอปพลิเคชันโทรศัพท์

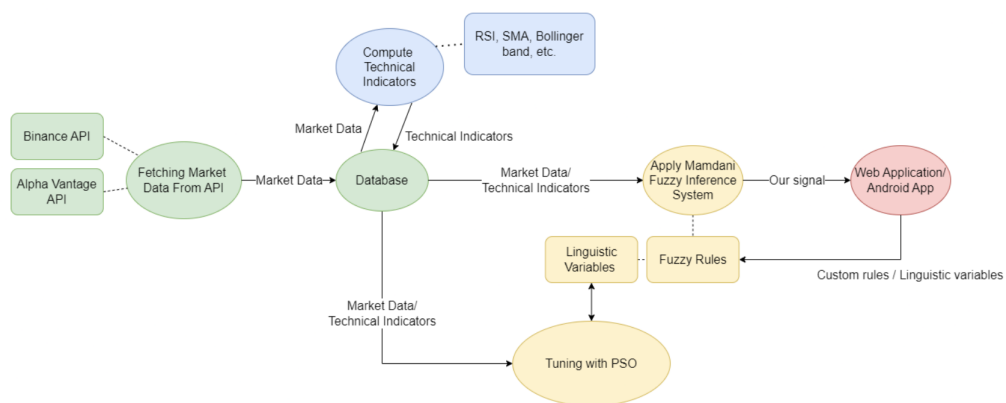
จุดประสงค์ของเว็บไซต์และแอปพลิเคชันคือเป็นส่วนติดต่อให้กับผู้ใช้งานที่ต้องการเข้ามาใช้ระบบของเราโดยมีส่วนที่ต้องรองรับหลักดังนี้

- ผู้ใช้งานสามารถดูกราฟ OHLC ของสินทรัพย์
- ผู้ใช้งานสามารถเพิ่มเครื่องมือตัวชี้วัดเบื้องต้นที่ต้องการอย่างเช่น RSI, MACD, และตัวอื่นๆ ที่ระบบของเรามีให้
- ผู้ใช้งานสามารถปรับแต่งระบบ Fuzzy logic (ปรับกฎ และตัวแปรทางภาษา)
- ผู้ใช้งานสามารถดูผลลัพธ์ที่ได้จากระบบ Fuzzy logic

ทำการออกแบบ UI/UX ของเว็บไซต์และแอปพลิเคชันโทรศัพท์ด้วย Figma โดยในการพัฒนาเว็บไซต์ส่วนหลักใช้ UI Framework อย่าง SvelteKit และภาษา TypeScript ส่วนของที่เป็นแอปพลิเคชันโทรศัพท์จะใช้งาน Flutter และภาษา Dart

3.7 แผนภาพกระแสข้อมูลโดยรวมของระบบ (Data Flow Diagram)

แผนภาพแสดงกระแสข้อมูลโดยเริ่มตั้งแต่การดึงข้อมูลตลาดจาก API มาเก็บที่ Database ซึ่งข้อมูลในนั้นจะถูกนำมาใช้งานคำนวณตัวชี้วัดทางเทคนิค, ประมวลผลและปรับตั้งระบบฟัซซี่ จนกระทั่งได้สัญญาณจากระบบฟัซซี่ไปแสดงบนเว็บไซต์และแอปพลิเคชันบนมือถือให้กับผู้ใช้งาน



รูปที่ 3.5: แผนภาพกระแสข้อมูล

บรรณานุกรม

- [1] Rodrigo Naranjo, Albert Meco, Javier Arroyo, and Matilde Santos. An intelligent trading system with fuzzy rules and fuzzy capital management. *International Journal of Intelligent Systems*, 30(8):963–983, 2015.
- [2] Alejandro Escobar, Julián Moreno, and Sebastián Múnera. A technical analysis indicator based on fuzzy logic. *Electronic Notes in Theoretical Computer Science*, 292:27–37, 2013. Proceedings of the XXXVIII Latin American Conference in Informatics (CLEI).
- [3] Ph.D.Associate Professor Sansanee Auephanwiriyaikul. *Introduction to Computational Intelligence for Computer Engineering*. 2013.
- [4] Ralph Vince. *Portfolio management formulas*. Wiley, 1990.

ภาคผนวก

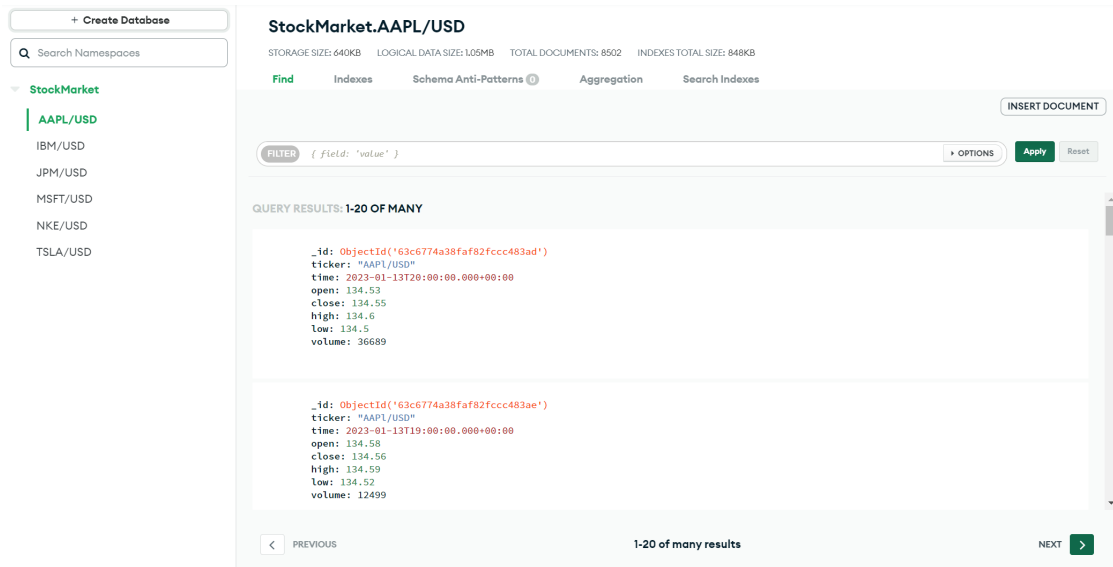
ภาคผนวก ก
ความคืบหน้าในการพัฒนา

Week	Work
4 Dec -10 Dec 2022	> Init Github repositories for both frontend and backend > Add basic implementation Fuzzy Logic from INTRO TO CI class
11 Dec -17 Dec 2022	
18 Dec - 24 Dec 2022	
25 Dec - 31 Dec 2022	
1 Jan - 7 Jan 2023	> Prepare contents for midterm presentation.
8 Jan - 14 Jan 2023	> Prepare contents for midterm presentation.
15 Jan - 21 Jan 2023	> Create MongoDB Instance > Insert stock market data to database using AlphaVantage
22 Jan - 28 Jan 2023	
29 Jan - 4 Feb 2023	> Searching about charting library for Frontend
5 Feb - 11 Feb 2023	
12 Feb - 18 Feb 2023	> Implementing a AWS lambda function for stock real-time updating > Designing Mobile Application's UI/UX > Start writing the report
19 Feb - 25 Feb 2023	> Designing Web Application's UI/UX
26 Feb - 4 Mar 2023	> Start refactoring our Fuzzy Logic library
5 Mar - 11 Mar 2023	> Refactoring most of our Fuzzy Logic library > Completed most of the report
12 Mar - 18 Mar 2023	> Refactor the remaining problem on our Fuzzy Logic library > Seed the crypto currency database, and implemented a AWS lambda function for crypto currency real-time updating > Fix the report

รูปที่ ก.1: ตารางการทำงาน

Backend

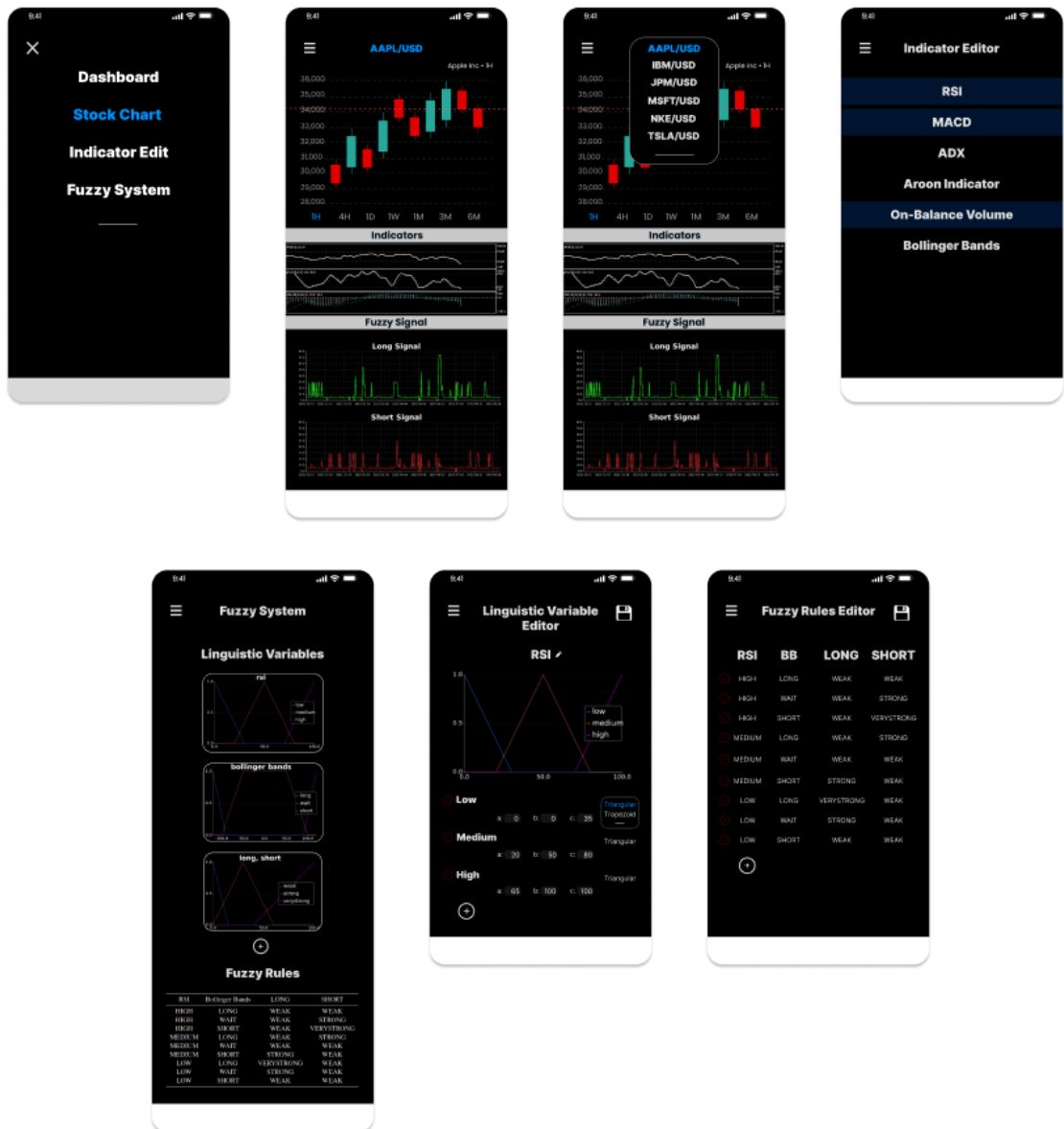
ได้มีการจัดเก็บข้อมูลตลาดหุ้น, ตลาด crypto currency และสร้างระบบอัปเดตข้อมูลอัตโนมัติ ได้เขียนโปรแกรมสำหรับ Fuzzy Logic ไปบางส่วนแล้ว รวมถึงมีการลองทำตัวเว็บเซิร์ฟเวอร์ไปบ้าง โดยสามารถดู code ได้ที่ <https://github.com/Fuzzy-Technical-Indicator/backend>



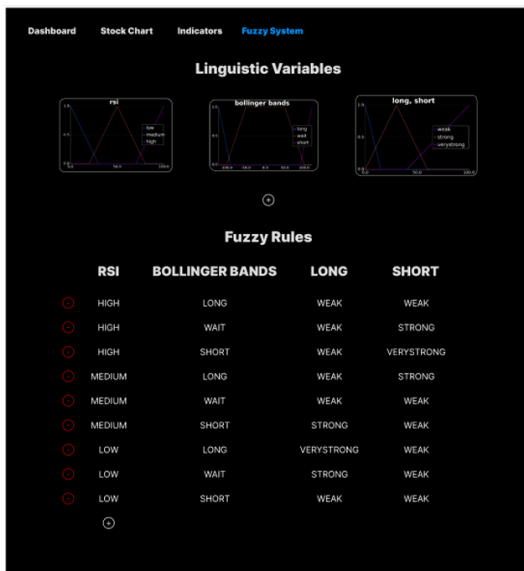
รูปที่ ก.2: ตัวอย่างข้อมูลตลาดหุ้นในฐานข้อมูล

Frontend

ได้มีการออกแบบหน้าตาแอปพลิเคชันทั้งแบบบนเว็บไซต์และแบบโทรศัพท์



รูปที่ ก.3: UI/UX ของแอปพลิเคชันบนโทรศัพท์



รูปที่ ก.4: UI/UX ของเว็บไซต์