อ่านกระดาน
Reading the Book

Part โบนัส: เครื่องจักรอ่านกระดาน — DeepLOB, Transformers

Order Flow & Microstructure สำหรับรายย่อย · เลนส์ execution ของซีรีส์

ML คือปลายทาง ไม่ใช่จุดเริ่ม — "feature ดี > โมเดลลึก" และอย่าทุ่ม architecture แทนข้อมูล

Part โบนัส — เครื่องจักรอ่านกระดาน

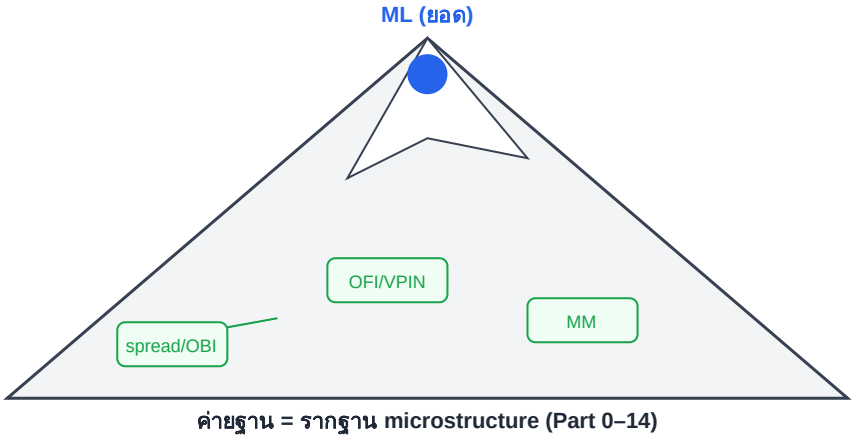
🎯 **อ่านจบ Part นี้คุณจะ**

- เข้าใจ pipeline ของโมเดล deep learning อ่านกระดาน: **LOB → CNN → LSTM → ทำนายทิศ mid** (DeepLOB)
- รู้จัก **Transformer สำหรับ LOB** (TLOB/LiT) และจุดที่มันต่างจาก CNN+LSTM
- ซึมหลักสำคัญ: **"feature ดี > โมเดลลึก"** — OFI/imbalance ที่ออกแบบดีชนะ network ใหญ่ที่ป้อนข้อมูลดิบ
- เข้าใจว่า **ML คือปลายทาง ไม่ใช่จุดเริ่ม** — ต้องมีรากฐาน microstructure ทั้ง 14 Part ก่อน

เริ่มจากยอดภูเขา ไม่ใช่ค่ายฐาน

คนที่พิชิตยอดเอเวอเรสต์ไม่ได้เริ่มจากการกระโดดขึ้นยอดเลย — เขาเริ่มจาก **ค่ายฐาน** ปรับตัวกับความสูง ฝึกหายใจ แล้วค่อยไต่ขึ้นทีละแคมป์ **ML สำหรับอ่านกระดานคือ "ยอดภูเขา"** — สวยงาม แต่ถ้าคุณไม่มีค่ายฐาน (เข้าใจ spread, OBI, OFI, VPIN, market making) คุณจะไม่รู้ด้วยซ้ำว่าโมเดลมันเรียนรู้อะไรหรือพังตรงไหน

อีกอุปมา: **สายตาเทรดเดอร์ที่ฝึกดูกระดานมาเป็นล้านครั้ง** — neural network ก็คือการจำลองสายตานั้นด้วยข้อมูลมหาศาล แต่ "สายตา" จะแม่นได้ต้องป้อน **สิ่งที่มีความหมาย** ให้ดู ไม่ใช่ตัวเลขดิบมั่ว ๆ

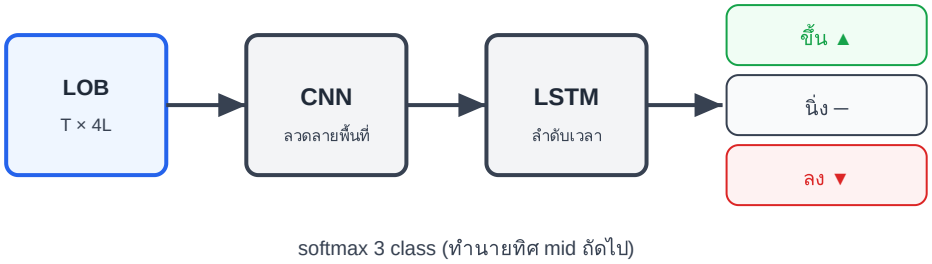


ภาพ B.1 · ML คือยอดภูเขา — ต้องมีค่ายฐาน microstructure ก่อนถึงจะไต่ขึ้นได้

ชั้นที่ 1 — pipeline DeepLOB (CNN + LSTM)

DeepLOB รับ **snapshot ของกระดานหลายชั้น ต่อกันเป็นลำดับเวลา** (เมทริกซ์ขนาด T × 4L: แต่ละชั้น L มี ราคา/ปริมาณ ของ bid และ ask) แล้วส่งผ่าน:

- **CNN** — กวาดหา "ลวดลายเชิงพื้นที่" ในกระดาน (เช่น รูปทรง imbalance ข้ามชั้น) แทนการดูตัวเลขทีละตัว
- **LSTM** — จับ "ลำดับเวลา" ว่าลวดลายเปลี่ยนยังไงในไม่กี่วินาที
- **softmax 3 class** — ทำนายว่า mid ช่วงถัดไปจะ ขึ้น / นิ่ง / ลง



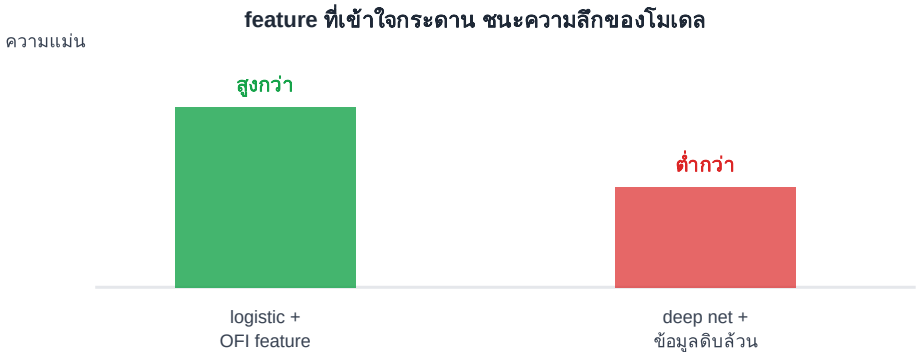
ภาพ B.2 · pipeline LOB → CNN → LSTM → ทำนาย 3 class

ชั้นที่ 2 — Transformer สำหรับ LOB (TLOB/LiT)

Transformer ใช้ **attention** แทน convolution + recurrence — มันเรียนได้เองว่า "ชั้นไหน/เวลาไหนของกระดาน" สำคัญต่อการทำนาย โดยไม่ต้องไล่ลำดับทีละสเต็ปแบบ LSTM งานอย่าง **TLOB** และ **LiT** แสดงว่าสถาปัตยกรรมนี้แข่งขันได้และบางทีดีกว่าในบางชุดข้อมูล — **แต่** มันกินข้อมูลและ compute หนักกว่ามาก และ *ไม่ใช่* กระสุนเงิน

ชั้นที่ 3 — feature ดี > โมเดลลึก (หัวใจของ Part นี้)

นี่คือบทเรียนที่แพงที่สุดและคนข้ามบ่อยที่สุด: **โมเดลลึกที่ป้อนข้อมูลดิบ มักแพ้โมเดลตื้นที่ป้อน feature ที่ออกแบบดี** เพราะ OFI, imbalance, microprice ที่คุณเรียนมาทั้งเล่มคือ "ความรู้ microstructure" ที่ย่อยมาให้โมเดลแล้ว — network ไม่ต้องเสียพลังไป "ค้นพบใหม่" จากศูนย์



ภาพ B.3 · feature ดี > โมเดลลึก — ความรู้ microstructure ที่ย่อยมาแล้วมีค่ากว่า depth

สูตรหลัก

```
// อินพุต: ลำดับ snapshot กระดาน T ช่วงเวลา × 4L ฟีเจอร์ต่อชั้น
X ∈ R^{T × 4L}  (p_bid,v_bid,p_ask,v_ask ของ L ชั้น)

// label: ทิศ mid ในอนาคต แบ่ง 3 class
y ∈ {ขึ้น, นิ่ง, ลง} (ใช้ threshold บนการเปลี่ยนของ mid)

// objective: ลด cross-entropy ของ softmax
min CrossEntropy( softmax(f(X)) , y )
   f = CNN+LSTM (DeepLOB) หรือ Transformer (TLOB/LiT)
```

⚠️ **ความจริงรายย่อย — กับดักของ ML กับกระดาน**

(1) ทุ่ม architecture แทน feature/ข้อมูล — เพิ่มเลเยอร์ไม่ช่วย ถ้า feature ห่วยหรือข้อมูลน้อย เริ่มที่ feature engineering ด้วยความรู้ microstructure ก่อนเสมอ **(2) ไม่กัน overfit** — LOB มี noise สูง และ regime เปลี่ยน โมเดลที่ "แม่นในอดีต" มักพังกับข้อมูลใหม่ ต้องแยก train/validation/test ตามเวลาอย่างเคร่งครัด และระวัง look-ahead bias **(3) คิดว่า ML แทนความเข้าใจได้** — ผิด ML ขยายความเข้าใจที่คุณมี ถ้าคุณไม่เข้าใจ OFI/imbalance คุณจะ debug โมเดลที่พังไม่เป็น **ML คือปลายทาง ไม่ใช่จุดเริ่ม**

📑 **งานวิจัยรองรับ**

DeepLOB (Zhang et al. 2018, arXiv 1808.03668) เป็นงานวางมาตรฐาน CNN+LSTM อ่าน LOB ทำนายทิศ mid ส่วน **TLOB (arXiv 2502.15757)** และ **LiT** นำ Transformer/attention มาใช้ และมีงานสำรวจ microstructure ในคริปโต LOB โดยเฉพาะ (**arXiv 2506.05764**) *ข้อควรระวัง:* ผลความแม่นในเปเปอร์มัก backtest บนข้อมูลเฉพาะตลาด/ช่วงเวลา — อย่าคาดหวังตัวเลขเดียวกันกับคริปโตที่ regime เปลี่ยนเร็ว และต้อง validate ด้วยข้อมูลของคุณเอง

🔗 เชื่อมโยงเล่มอื่น & ปิดซีรีส์

Part โบนัสมัดทั้งซีรีส์เข้าด้วยกัน: **feature** ที่ป้อนโมเดลคือ OFI (Part 7), OBI (Part 5), microprice (Part 11), VPIN (Part 8) ที่เรียนมาทั้งหมด ในเชิง **math** นี่คือ softmax/cross-entropy, logistic regression, และ optimization ที่ต่อยอดจากบท regression/PCA โดยตรง บทเรียนสุดท้ายของทั้งเล่ม "อ่านกระดาน" คือ: **order book เป็นเครื่องมือยกระดับกลยุทธ์เดิม** ไม่ใช่ระบบเบ็ดเสร็จ — และ ML ก็เป็นเครื่องมือยกระดับ "ความเข้าใจกระดาน" ของคุณ ไม่ใช่ตัวแทนของมัน

✍️ แบบฝึก (ทำได้ด้วยข้อมูลคริปโตฟรี)

เก็บข้อมูล L2 + trade ของ BTC/USDT สัก 2–3 ชั่วโมง สร้าง feature **OFI** (จาก Part 7) แล้วเทรน **logistic regression** ทำนายทิศ mid ถัดไป 3 class เทียบกับ baseline ง่าย ๆ คือ "เดาตาม sign(OFI)" ดูว่าโมเดลที่เรียนรู้ค่าน้ำหนักชนะการเดาตรง ๆ ไหม จากนั้น (ถ้าอยากท้าทาย) ลองป้อนข้อมูลกระดานดิบเข้า network เล็ก ๆ แล้วเทียบ — คุณจะได้สัมผัสด้วยตัวเองว่า "feature ดี > โมเดลลึก" จริงแค่ไหน

— จบซีรีส์ "อ่านกระดาน" · ขอให้อ่านกระดานออก และเทรดอย่างมีต้นทุนที่รู้จริง —