1. ทำไมต้อง XAI (Explainable AI)

AI สมัยใหม่ โดยเฉพาะโมเดลที่ซับซ้อนอย่าง Deep Learning นั้นเก่งมากในการทำนายผลลัพธ์ แต่บ่อยครั้งมันทำงานเหมือน "กล่องดำ" (Black Box) คือมันให้คำตอบเรา แต่ไม่ยอมบอกว่าทำไมถึงได้คำตอบนั้น

XAI จึงเกิดขึ้นมาเพื่อสร้าง "คำอธิบาย" ให้เราเข้าใจได้ว่า มีปัจจัยอะไรบ้างที่ผลักดันให้ AI ตัดสินใจแบบนั้น สิ่งนี้ช่วยสร้างความน่าเชื่อถือ และช่วยให้เรารู้จุดที่ต้องแก้ไขเมื่อ AI ทำงานผิดพลาด

2. Interpretability vs Explainability

- **Interpretability (การตีความได้):** หมายถึง โมเดลที่ "โปร่งใส" โดยธรรมชาติ [9]เราสามารถอ่านโครงสร้างของมันแล้วเข้าใจวิธีคิดได้ทันที เช่น โมเดล Linear Regression หรือ Decision Tree ในไฟล์เปรียบเปรยว่าเหมือน "อ่านป้ายก็รู้เรื่อง"
- **Explainability (การอธิบายได้):** ใช้กับโมเดล Black Box ที่โครงสร้างซับซ้อนเกินกว่ามนุษย์จะอ่านเข้าใจ เราจึงต้องใช้ "เครื่องมือ" หรือเทคนิคอื่นมาช่วยอธิบายการตัดสินใจของมันทีหลัง เหมือน "ต้องมีล่ามมาแปล" ให้เราฟังอีกที

3. Global vs Local Explanation

- **Global (ภาพรวม):** อธิบายพฤติกรรมโดยรวมของโมเดล เช่น "สำหรับโมเดลนี้ ปัจจัยเรื่อง 'รายได้' สำคัญที่สุดในการอนุมัติสินเชื่อ" นี่คือการมองภาพใหญ่ว่าฟีเจอร์ไหนสำคัญโดยเฉลี่ย
- **Local (รายกรณี):** อธิบายการตัดสินใจสำหรับข้อมูล แค่จุดเดียว เช่น "ทำไม นาย A ถึงไม่ผ่านสินเชื่อ?" คำตอบอาจจะเป็น "เพราะนาย A มีหนี้สูง แม้รายได้จะเยอะก็ตาม" นี่คือการเจาะลึกเฉพาะเคส

4. โมเดลตีความได้เอง vs อธิบายทีหลัง

นี่คือการแบ่งประเภทโมเดลตามข้อ 2

- **โมเดลตีความได้เอง (Interpretable Models):** เช่น Linear / Logistic Regression, Decision Tree, หรือการสร้างกฎ If-Then โมเดลเหล่านี้อธิบายตัวเองได้ตรงๆ
- **โมเดลที่ต้องอธิบายทีหลัง (Post-hoc Explainability):** คือโมเดล Black Box ที่ซับซ้อน เช่น Random Forest, Boosting อย่าง Neural Networks ที่ต้องใช้ เครื่องมือ อย่าง LIME หรือ SHAP มาช่วยอ่านการตัดสินใจของมัน

5. PDP (Partial Dependence Plot)

PDP เป็นเทคนิคอธิบายแบบ Global มันช่วยให้เราเห็นความสัมพันธ์ระหว่างฟีเจอร์ 1 ตัว กับผลลัพธ์ "โดยเฉลี่ย" ของข้อมูลทั้งหมด

ตัวอย่าง: ถ้าเราอยากรู้ว่า "อุณหภูมิ" ส่งผลต่อ "ยอดเช่าจักรยาน" ยังไง

PDP จะคำนวณโดยการ "สมมติ" ให้ทุกเคสในข้อมูลมีอุณหภูมิเท่ากัน (เช่น 20 องศา) แล้วดูว่าค่าเฉลี่ยยอดเช่าเป็นเท่าไหร่ จากนั้นอุณหภูมิให้สูงขึ้นไปเรื่อยๆ แล้วพล็อตกราฟ ผลลัพธ์อาจจะเห็นเป็นเส้นโค้งว่า ยิ่งอุณหภูมิสูง ยอดเช่าเฉลี่ยก็สูงขึ้น

6. ICE (Individual Conditional Expectation)

ICE คล้าย PDP แต่เป็นเวอร์ชัน Local

แทนที่จะดู "ค่าเฉลี่ย" ของทุกเคสเหมือน PDP, ICE จะวาดเส้นกราฟความสัมพันธ์นั้น แยกให้เห็นทีละเคส (ต่อคน/ต่อวัน) สิ่งนี้มีประโยชน์มาก เพราะ "ค่าเฉลี่ย" อาจซ่อนพฤติกรรมที่แตกต่างกันไว้

ตัวอย่าง: PDP อาจบอกว่าอุณหภูมิสูงขึ้น ยอดเช่าเฉลี่ย สูงขึ้น แต่ ICE อาจแสดงให้เห็นว่า มีคนบางกลุ่ม เช่น คนแพ้อากาศร้อน ที่ยอดเช่า ลดลง สวนทางกับค่าเฉลี่ย

7. LIME (Local Interpretable Model-agnostic Explanations)

LIME ใช้อธิบาย Black Box แบบ Local (รายเคส)

หลักการทำงาน: สมมติเราอยากรู้ว่าทำไมเคส "นาย A" ถึงถูกทำนายว่า "เสี่ยงสูง"

1. LIME จะสร้างข้อมูล "ปลอม" ขึ้นมาหลายๆ แบบที่ "คล้ายๆ กับนาย A" เช่น อายุเท่ากันแต่เงินเดือนต่างกันเล็กน้อย, เงินเดือนเท่ากันแต่อายุต่างกันเล็กน้อย
2. LIME จะส่งข้อมูลปลอมเหล่านี้ไปถามโมเดล Black Box ว่าได้ผลลัพธ์อะไร
3. LIME นำผลลัพธ์ที่ได้ มาสร้าง "โมเดลง่ายๆ" เช่น Linear Regression ที่อธิบายพฤติกรรมของ Black Box เฉพาะบริเวณใกล้ๆ เคสนาย A
4. โมเดลง่ายๆ นี้จะบอกเราว่า "สำหรับเคสที่คล้ายๆ นาย A เนี่ย ปัจจัยที่ 'ดัน' ให้เสี่ยงสูงคือ 'หนี้เยอะ' และปัจจัยที่ 'ดึง' ให้เสี่ยงต่ำคือ 'อายุงานนาน'"

8. SHAP (SHapley Additive exPlanations)

SHAP ใช้อธิบายได้ทั้ง Local และ Global

หลักการทำงาน: SHAP มาจากทฤษฎีเกม แนวคิดคือการ "แบ่งเครดิต" หรือ "แบ่งแต้ม" อย่างยุติธรรม ให้กับฟีเจอร์แต่ละตัว ว่ามีส่วนช่วย "ผลัก" หรือ "ดึง" ผลการทำนายมากน้อยแค่ไหน

- **Local:** อธิบายเคสนาย A ว่า "นาย A ได้คะแนนความเสี่ยง 0.8 เพราะ:
    - ปัจจัยฐาน (ค่าเฉลี่ย) เริ่มที่ 0.5
    - 'เงินเดือนน้อย' ผลักขึ้น +0.1
    - 'หนี้สูง' ผลักขึ้น +0.3
    - 'อายุยังน้อย' ผลักลง -0.1
- **Global:** เมื่อเรามีแต้มของทุกเคสแล้ว เราสามารถนำมาสรุปเป็นภาพรวมได้ว่า ฟีเจอร์ไหนมีอิทธิพลต่อโมเดลมากที่สุดโดยเฉลี่ย

9. Grad-CAM (สำหรับรูปภาพ)

Grad-CAM เป็นเทคนิคสำหรับโมเดลที่ประมวลผล "รูปภาพ" เช่น Neural Networks มันจะสร้าง "แผนที่ความร้อน" ซ้อนทับลงบนรูปภาพต้นฉบับ เพื่อแสดงว่า "โมเดลกำลังมองส่วนไหนของภาพ" ตอนที่มันตัดสินใจ **ตัวอย่าง:** ถ้าเราสั่งให้โมเดลทายภาพ "แมว"

- **ดี:** Heatmap ขึ้นสีเข้มที่บริเวณใบหน้า หู และตาของแมว แปลว่าโมเดลมองถูกจุด
- **ไม่ดี:** Heatmap ไปขึ้นสีเข้มที่ "ฉากหลัง" หรือ "มุมพื้น" แปลว่าโมเดลอาจจะไม่ได้รู้จักแมวจริงๆ แต่อาจจะแค่จำได้ว่า "ถ้าเจอพื้นลายนี้ มักจะเป็นแมว" ซึ่งเป็นวิธีคิดที่ผิดและไม่น่าเชื่อถือ

10. Counterfactual Explanation (คำอธิบายเชิงหักล้าง)

นี่เป็นคำอธิบายที่ "ปฏิบัติได้จริง" แทนที่จะบอกว่า ทำไม คุณถึงได้ผลลัพธ์นี้ มันจะบอกว่า "คุณต้องเปลี่ยนแปลงอะไร เล็กน้อย ที่สุด เพื่อให้ผลลัพธ์ เปลี่ยน เป็นอีกแบบ"

ตัวอย่าง:

- **คำอธิบายปกติ (เช่น LIME):** "คุณไม่ผ่านสินเชื่อเพราะเงินเดือนน้อยและหนี้สูง"
- **คำอธิบาย Counterfactual:** "ถ้าเงินเดือนของคุณเพิ่มขึ้นอีก 3,000 บาท โดยที่ปัจจัยอื่นเท่าเดิม คุณจะผ่านการอนุมัติสินเชื่อ" เป็นการบอก "แนวทางแก้ไข" ที่ชัดเจนให้ผู้ใช้

11. Logistic Regression (โมเดลตีความได้เอง)

เป็นโมเดลคลาสสิกที่ "โปร่งใส" หลักการคือ มันจะให้ "คะแนน" กับแต่ละปัจจัย คะแนนอาจเป็นบวก (ช่วยเพิ่มโอกาส) หรือลบ (ช่วยลดโอกาส)

ตัวอย่าง: ทำนายการอนุมัติสินเชื่อ

- เงินเดือน (ทุก 10,000 บาท): +0.5 คะแนน
- หนี้สิน (ทุก 10,000 บาท): -1.0 คะแนน
- อายุงาน (ทุก 1 ปี): +0.2 คะแนน

โมเดลจะ "รวมคะแนน" ทั้งหมดของคนๆ นั้น แล้วแปลงคะแนนรวมให้เป็น "ความน่าจะเป็น" (ค่าระหว่าง 0-1) เช่น ถ้าคะแนนรวมสูงมาก อาจจะได้ 0.95 โอกาสผ่าน 95% ถ้าคะแนนติดลบ อาจจะได้ 0.10 โอกาสผ่าน 10%

12. Decision Tree (โมเดลตีความได้เอง)

เป็นโมเดลที่ทำงานเหมือนการเล่นเกม ถาม-ตอบ 20 คำถาม

มันจะสร้างชุดคำถาม "ใช่/ไม่ใช่" ต่อกันไปเรื่อยๆ เป็นโครงสร้างเหมือนต้นไม้

เราแค่เดินตามคำตอบของเคสนั้นๆ ไปทีละกิ่ง จนถึง "ใบ" สุดท้าย ก็จะได้คำทำนาย เราสามารถย้อนกลับไปดูเส้นทางที่เดินมาเพื่ออธิบายได้ทันทีว่าทำไมถึงได้คำตอบนี้

13. EBM (Explainable Boosting Machine)

EBM เป็นโมเดลที่ทำงานโดยการสร้าง "เส้นโค้งผลกระทบ" แยกสำหรับแต่ละฟีเจอร์ เช่น

- เส้นโค้งของ "อายุ" อาจจะบอกว่า ความเสี่ยงจะต่ำตอนหนุ่มสาว, ค่อยๆ สูงขึ้นตอนกลางคน, และคงที่ตอนสูงอายุ
- เส้นโค้งของ "รายได้" อาจจะบอกว่า ความเสี่ยงจะสูงมากตอนรายได้น้อย, และลดลงฮวบเดียวเมื่อรายได้เกิน 20,000

เวลาทำนาย EBM จะแค่ "รวมผลกระทบ" จากทุกเส้นโค้งเข้าด้วยกัน ทำให้เราสามารถดูกราฟของแต่ละฟีเจอร์เพื่อเข้าใจผลกระทบของมันได้ชัดเจน

14. Forest-Guided Clustering

เทคนิคนี้ใช้ประโยชน์จากโมเดล Random Forest ซึ่งเป็น Black Box เพื่อมาช่วย "จัดกลุ่ม" ข้อมูล Random Forest ประกอบด้วย Decision Trees หลายร้อยต้น

แนวคิดคือ "ถ้าข้อมูล 2 ชิ้น (เช่น คน 2 คน) ตกไปอยู่ใน 'ใบไม้' สุดท้ายใบเดียวกันบ่อยๆ ในหลายๆ ต้นไม้ แสดงว่าคน 2 คนนี้มีความคล้ายกันมากในสายตาของโมเดล" เราจึงใช้โครงสร้างนี้มาจับกลุ่มคนที่โมเดล "คิด" ว่าเหมือนกัน แล้วมาสรุปว่าแต่ละกลุ่มมี ฟีเจอร์เด่น อะไร ทำให้เห็นภาพรวมว่าโมเดลแบ่งแยกผู้คนด้วยเหตุผลแบบไหน

15. ทำไมต้อง XAI ใน NLP (งานด้านภาษา/ข้อความ)

โมเดลภาษา (NLP) ต้องอ่าน "คำ" หลายๆ คำในประโยค เราจึงอยากรู้ว่า "คำไหน" หรือ "กลุ่มคำไหน" ในประโยค ที่ทำให้โมเดลตัดสินใจแบบนั้น เพื่อตรวจสอบว่ามันเข้าใจถูกจุดหรือไม่

ตัวอย่าง: ประโยค "หนังเรื่องนี้ภาพสวยมาก แต่เนื้อเรื่องน่าเบื่อสุดๆ" -> โมเดลทายว่า "แง่ลบ"

- **LIME สำหรับข้อความ:** จะทำงานโดยการ "ลองปิดคำ" เช่น ลองลบคำว่า "น่าเบื่อ" ออก แล้วดูว่าคำตอบเปลี่ยนไหม ถ้าลบแล้วคำตอบเปลี่ยน เช่น กลายเป็น "แง่บวก" แสดงว่าคำว่า "น่าเบื่อ" มีอิทธิพลสูงมาก
- **SHAP สำหรับข้อความ:** จะ "แบ่งแต้ม" ให้แต่ละคำ
    - "ภาพสวย"+0.3 ดันไปทางบวก / "น่าเบื่อ"-0.6 ดันไปทางลบ / "สุดๆ"-0.2 ดันไปทางลบ
    - รวมแต้มแล้วติดลบ จึงทายว่า "แง่ลบ" โดยมีคำว่า "น่าเบื่อ" เป็นตัวหลัก

16. Gradient-based Explanations (คำอธิบายฐาน Gradient)

นี่เป็นเทคนิคสำหรับ Neural Networks คือ

"Gradient" คือค่าที่บอกว่า "ถ้าเราเปลี่ยน Input เช่น คำ นิดเดียว Output ที่เป็นคำทำนาย จะเปลี่ยนไปแค่ไหน"

เทคนิคนี้จะดู "ลำดับ" ของข้อมูลด้วย แทนที่จะมองแค่ "คำนี้" เหมือน LIME/SHAP มันอาจจะมองว่า "คำนี้ตามด้วย คำนี้" ส่งผลต่อโมเดลอย่างไร ผลลัพธ์มักจะออกมาเป็น “Saliency Map” แผนที่ความเด่น ซึ่งคล้ายๆ Heatmap คือการไฮไลต์ว่าคำไหน ที่มีค่า Gradient สูงสุด คืออ่อนไหวต่อการเปลี่ยนแปลงมากที่สุด

17. Attention (กลไกการโฟกัส)

Attention เป็น "ชิ้นส่วน" ที่อยู่ ข้างใน โมเดลตระกูล Transformer เช่น GPT มันถูกออกแบบมาให้โมเดล "โฟกัส" หรือ "ให้ความสนใจ" กับคำที่เกี่ยวข้องในประโยค

ตัวอย่าง: ในประโยค "แมวกินปลา" ตอนที่โมเดลประมวลผลคำว่า "กิน" กลไก Attention อาจจะช่วยให้มันโฟกัสไปที่ "แมว" (ใครกิน) และ "ปลา" (กินอะไร)

เราสามารถดึงค่า Attention นี้ออกมาดูได้ ว่าโมเดลโฟกัสตรงไหน

แต่ Attention ไม่ได้เท่ากับ 'ความหมาย' หรือ 'คำอธิบาย' เสมอไป บางครั้งโมเดลก็แค่โฟกัสคำที่มีความถี่สูง หรือคำทางเทคนิคบางอย่าง ไม่ได้แปลว่าคำนั้นสำคัญที่สุดต่อความหมายเสมอไป

18. บทเรียนด้านจริยธรรม (Ethics) และ ความยุติธรรม (Fairness)

นี่คือหัวข้อที่สำคัญที่สุด XAI ไม่ใช่แค่เรื่องเทคนิค แต่เป็นเรื่องความรับผิดชอบ

1. **เก็บข้อมูลให้ครบ:** ต้องมั่นใจว่าข้อมูลที่ใช้สอน AI มาจากทุกกลุ่มประชากร ไม่เอนเอียงไปกลุ่มใดกลุ่มหนึ่ง
2. **ระวังตัวแปรแฝง (Proxy):** ห้ามใช้ฟีเจอร์ที่อาจนำไปสู่การเลือกปฏิบัติทางอ้อม เช่น การใช้ "รหัสไปรษณีย์" ซึ่งในบางพื้นที่อาจจะเชื่อมโยงกับ "เชื้อชาติ" หรือ "ฐานะ" อย่างชัดเจน
3. **วัดผลให้รอบด้าน:** อย่าดูแค่ "ความแม่นยำรวม" (Accuracy) ต้องเจาะลึกด้วยว่า "โมเดลนี้แม่นยำเท่ากัน ในทุกกลุ่มหรือไม่?" เช่น แม่นกับผู้ชาย 90% แต่แม่นกับผู้หญิงแค่ 60% ถือว่าไม่ยุติธรรม
4. **อธิบายและแก้ไขได้:** ต้องอธิบายผลให้ผู้ใช้เข้าใจง่าย และมีช่องทางให้ผู้ใช้สามารถอุทธรณ์หรือแก้ไขการตัดสินใจที่ผิดพลาดได้