การจำแนกทักษะของวิศวกรซอฟต์แวร์สำหรับสร้างตัวแบบสมรรถนะตามความต้องการขององค์กรซอฟต์แวร์ในประเทศไทย

Classification of software engineering skills for competency model with the demands of Thailand's software organizations

นางสาวณัฐพร ภักดี

นายอริย์ธัช ศิรภัทร์วงศ์กร

โครงการวิจัยประเภทเงินรายได้

คณะวิทยาการสารสนเทศ

ประจำปีงบประมาณ พ.ศ. 2567

มหาวิทยาลัยบูรพา

เลขที่สัญญา 6/2567

รายงานวิจัยฉบับสมบูรณ์

โครงการ การจำแนกทักษะของวิศวกรซอฟต์แวร์สำหรับพัฒนาตัวแบบสมรรถนะตามความต้องการขององค์กรซอฟต์แวร์ในประเทศไทย

(Classification of software engineering skills for competency model with the demands of Thailand's software organizations)

ณัฐพร ภักดี
อริย์ธัช ศิรภัทร์วงศ์กร

คณะวิทยาการสารสนเทศ
กรกฎาคม พ.ศ. 2568

กิตติกรรมประกาศ

รายงานวิจัยนี้ได้รับทุนสนับสนุนการวิจัยจากเงินรายได้ คณะวิทยาการสารสนเทศ ประจำปีงบประมาณ 2567 เลขที่สัญญา 6/2567

ผู้วิจัยขอขอบพระคุณ คณะวิทยาการสารสนเทศ มหาวิทยาลัยบูรพา ที่ได้ให้โอกาสในการทำวิจัย รวมถึงเงินทุนเพื่อส่งเสริมและพัฒนาศักยภาพนักวิจัย มา ณ โอกาสนี้

ณัฐพร ภักดี

กรกฎาคม 2568

บทคัดย่อภาษาไทย

งานวิจัยนี้มีวัตถุประสงค์เพื่อวิเคราะห์ทักษะที่เป็นที่ต้องการในอุตสาหกรรมซอฟต์แวร์ของประเทศไทย โดยมุ่งเน้นทั้งทักษะทางเทคนิค (Technical Skills) และทักษะด้านบุคคล (Soft Skills) ที่ปรากฏในประกาศรับสมัครงาน โดยใช้เทคนิคการประมวลผลภาษาธรรมชาติ (Natural Language Processing: NLP) เพื่อดึงข้อมูลเชิงลึกจากข้อความประกาศงาน ข้อมูลถูกเก็บจากเว็บไซต์ JobThai และ LinkedIn ในช่วงปี 2023–2024 จากนั้นผ่านกระบวนการทำความสะอาดข้อมูล การตัดคำ การลบคำซ้ำ และการวิเคราะห์ความถี่ของทักษะในแต่ละตำแหน่งงาน ผลการศึกษาพบว่า ทักษะด้าน Programming Languages เช่น Java, SQL และ Python เป็นที่ต้องการสูงที่สุดในเกือบทุกตำแหน่ง ขณะที่ทักษะด้านบุคคล เช่น Communication, Collaborate และ Problem-Solving เป็นทักษะหลักที่ปรากฏอย่างสม่ำเสมอในทุกตำแหน่งเช่นกัน

นอกจากนี้ยังมีการเปรียบเทียบประสิทธิภาพของโมเดลการทำนายตำแหน่งงานโดยใช้เทคนิค Machine Learning ได้แก่ Random Forest, Naïve Bayes และ XGBoost และ Deep Learning ได้แก่ LSTM ผลลัพธ์พบว่า Random Forest ให้ผลดีที่สุด ผลที่ได้จากงานวิจัยนี้สามารถนำไปใช้เป็นข้อมูลสำหรับการปรับปรุงหลักสูตรการเรียนการสอนให้ตอบโจทย์ตลาดแรงงาน และการใช้โมเดลเชิงพยากรณ์เพื่อแนะนำทักษะที่เหมาะสมในอนาคต

ABSTRACT

This research aims to analyze the in-demand skills within Thailand's software industry, focusing on both technical skills and soft skills as found in real job postings. Natural Language Processing (NLP) techniques were applied to extract insights from job descriptions. Data were collected from JobThai and LinkedIn, totaling records during 2023–2024. The dataset underwent cleaning, tokenization, deduplication, and frequency analysis to identify dominant skills across different job roles. The findings revealed that programming languages such as Java, SQL, and Python were the most frequently required technical skills across nearly all positions. On the soft skill side, communication, collaboration, and problem-solving consistently ranked as the top three skills in every job category.

Furthermore, the study compared the performance of various machine learning models for job role prediction, including Random Forest, Naïve Bayes, and XGBoost, as well as a deep learning model (LSTM). Among these, Random Forest achieved the highest performance. The results of this study can be used to inform curriculum development aligned with labor market needs and support the use of predictive models to recommend future skill development strategies.

สารบัญ

หน้า

บทคัดย่อภาษาไทย ..... ก

ABSTRACT ..... ข

สารบัญ ..... ค

สารบัญตาราง ..... จ

สารบัญภาพ ..... ฉ

บทที่ 1 บทนำ 1

1.1 ความเป็นมาและความสำคัญของปัญหา 1

1.2 วัตถุประสงค์งานวิจัย 2

1.3 ขอบเขตและกรอบแนวคิดการวิจัย 2

1.4 นิยามศัพท์เฉพาะ 3

1.5 ประโยชน์ที่ได้รับจากงานวิจัย 4

บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง 5

2.1 สมรรถนะ (Competence) 5

2.2 การวิเคราะห์ข้อความ (Text Mining) 8

2.3 การจัดการกับข้อมูลที่ไม่สมดุล (SMOTE) 16

2.4 เทคนิคการจำแนกกลุ่มข้อมูล (Classification) 16

2.5 การวัดประสิทธิภาพของโมเดล 21

2.6 งานวิจัยที่เกี่ยวข้อง 23

บทที่ 3 วิธีการดำเนินการวิจัย 26

3.1 ศึกษาสมรรถนะตามเอกสารและงานวิจัยที่เกี่ยวข้องกับวิศวกรรมซอฟต์แวร์ 26

3.2 ศึกษาสมรรถนะของวิศวกรรมซอฟต์แวร์จากประกาศรับสมัครงาน 27

หน้า

บทที่ 4 ผลการดำเนินการวิจัย 35
4.1 ผลการวิเคราะห์ความถี่และร้อยละของสมรรถนะที่ได้จากแหล่งข้อมูลประกาศรับสมัครงาน ... 35
4.2 ผลการวิเคราะห์ชุดทักษะตามความต้องการในแต่ละตำแหน่งงาน 48
4.3 ตัวแบบการทำนายตำแหน่งงานตามกลุ่มทักษะตามอาชีพด้วยเทคนิคการจำแนกประเภท 74
บทที่ 5 สรุปและอธิปรายผล 75
5.1 สรุปผลการวิจัย 75
5.2 ข้อเสนอแนะงานวิจัย 77
5.3 ข้อจำกัดของงานวิจัย 77
เอกสารอ้างอิง 78

สารบัญตาราง

ตารางที่ หน้า

1-1 กลุ่มคำของทักษะด้านบุคคล (Soft skill)..... 3
2-1 ประเภทของสมรรถนะ..... 6
2-2 กรอบแนวคิด Hard Skills และ Soft Skills..... 7
3-1 ตัวอย่างคำที่ไม่เกี่ยวข้อง (Remove word)..... 29
3-2 ตัวอย่างการจัดการข้อความที่มีความสอดคล้องกัน..... 29
3-3 ทักษะด้านเทคนิค (Technical skill)..... 30
3-4 กลุ่มคำของทักษะด้านบุคคล (Soft skill)..... 31
3-5 จำนวนประกาศรับสมัครงานก่อนและหลังการปรับโมเดลด้วย SMOTE..... 32
4-1 จำนวนประกาศรับสมัครงานจำแนกตามตำแหน่งงาน..... 36
4-2 จำนวนและร้อยละทักษะทั้งหมดประกาศรับสมัครงานจำแนกทักษะเชิงเทคนิค..... 36
4-3 จำนวนและร้อยละของทักษะด้านภาษาโปรแกรม..... 37
4-4 จำนวนและร้อยละของทักษะด้าน Frameworks & Libraries..... 38
4-5 จำนวนและร้อยละของทักษะด้าน Cloud & Infrastructure..... 39
4-6 จำนวนและร้อยละของทักษะด้าน Database & Data Technologies..... 40
4-7 จำนวนและร้อยละของทักษะด้าน Software Architecture & Design..... 41
4-8 จำนวนและร้อยละของทักษะด้าน Development Tools..... 42
4-9 จำนวนและร้อยละของทักษะด้าน AI/ML/Data Science..... 43
4-10 จำนวนและร้อยละของทักษะด้าน Testing Tools & Automation..... 44
4-11 จำนวนและร้อยละของทักษะด้าน Security Tools & Techniques..... 45
4-12 จำนวนและร้อยละของทักษะด้าน Mobile Development..... 46
4-13 จำนวนและร้อยละทักษะทั้งหมดประกาศรับสมัครงานจำแนกทักษะด้านบุคคล..... 47
4-14 ผลลัพธ์การเปรียบเทียบประสิทธิภาพของโมเดล..... 74

สารบัญภาพ

ภาพที่ หน้า

2-1 กรอบแนวคิดของ Zhang (2012) 7
2-2 กระบวนการค้นหาลักษณะแฝงของข้อมูล (Crisp DM) 9
2-3 ขั้นตอนการสร้างกฎความสัมพันธ์ 12
2-4 Decision tree Structure 17
2-5 Random Forest Structure 18
2-6 การจำลองกลุ่มข้อมูลด้วยเทคนิค XGBoost 19
3-1 การออกแบบกระบวนการวิจัย 34
4-1 สัดส่วนทักษะด้านเทคนิคของวิศวกรความต้องการ (Requirements Engineer) 49
4-2 ทักษะเชิงเทคนิคของตำแหน่งวิศวกรความต้องการ(Requirements Engineer) 50
4-3 ทักษะเชิงบุคคล 10 ลำดับแรกของตำแหน่งวิศวกรความต้องการ 51
4-4 สัดส่วนทักษะด้านเทคนิคของตำแหน่งนักออกแบบซอฟต์แวร์ 52
4-5 ทักษะเชิงเทคนิคของตำแหน่งนักออกแบบซอฟต์แวร์ 53
4-6 ทักษะเชิงเทคนิคของตำแหน่งนักออกแบบซอฟต์แวร์ 54
4-7 สัดส่วนทักษะด้านเทคนิคของนักพัฒนาซอฟต์แวร์ 55
4-8 ทักษะเชิงเทคนิคของตำแหน่งนักพัฒนาซอฟต์แวร์ 56
4-9 ทักษะเชิงบุคคลของตำแหน่งนักพัฒนาซอฟต์แวร์ 57
4-10 สัดส่วนทักษะด้านเทคนิคของนักทดสอบซอฟต์แวร์ 58
4-11 ทักษะเชิงเทคนิคของตำแหน่งนักทดสอบซอฟต์แวร์ 59
4-12 ทักษะเชิงบุคคล 10 ลำดับแรกของตำแหน่งนักทดสอบซอฟต์แวร์ 60
4-13 สัดส่วนทักษะด้านเทคนิคของวิศวกรซ่อมบำรุงซอฟต์แวร์ 62
4-14 ทักษะเชิงเทคนิคของตำแหน่งวิศวกรซ่อมบำรุงซอฟต์แวร์ 63
4-15 ทักษะเชิงบุคคลของตำแหน่งวิศวกรซ่อมบำรุงซอฟต์แวร์ 64
4-16 สัดส่วนทักษะด้านเทคนิคของดการโครงการซอฟต์แวร์ 65
4-17 ทักษะเชิงเทคนิคของตำแหน่งผู้จัดการโครงการซอฟต์แวร์ 67
4-18 ทักษะเชิงบุคคล 10 อันดับแรกของตำแหน่งผู้จัดการโครงการ 68
4-19 สัดส่วนทักษะด้านเทคนิคของตำแหน่งนักวิเคราะห์ระบบ 70
4-20 ทักษะด้านเทคนิคของตำแหน่งนักวิเคราะห์ระบบ 71
4-21 ทักษะด้านบุคคลของตำแหน่งนักวิเคราะห์ระบบ 73

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ในการเปลี่ยนแปลงและแข่งขันของธุรกิจทุกภาคส่วนเมื่อเกิดการแพร่ระบาดครั้งใหญ่ของ Covid -19 ทำให้อุตสาหกรรมซอฟต์แวร์ในประเทศไทยมีการเติบโตมากขึ้น ทำให้องค์กรต่างๆ ต้องพึ่งพาเทคโนโลยีสารสนเทศในการดำเนินธุรกิจ [1-2] โดยซอฟต์แวร์เป็นส่วนสำคัญของเทคโนโลยีทุกภาคส่วนไม่ว่าจะเป็นด้านสุขภาพ การศึกษา การเงิน การขนส่ง ความบันเทิง เป็นต้น จึงทำให้ความต้องการบุคลากรในการพัฒนาซอฟต์แวร์มีมากขึ้น เหตุผลที่ความต้องการเพิ่มขึ้นนี้มาจากบริษัทต้องการพัฒนาเว็บไซต์และเว็บแอปพลิเคชันเพื่อสนองตอบความต้องการของลูกค้า รวมถึงจำนวนผลิตภัณฑ์ที่ใช้ซอฟต์แวร์ต่างๆ จำเป็นต้องมีการปรับปรุงโค้ดเพื่อให้สามารถใช้งานได้และสอดคล้องกับเทคโนโลยีที่เกิดขึ้นใหม่ [3] กระทรวงดิจิทัลและเทคโนโลยีประเทศไทย มีการสำรวจตลาดอุตสาหกรรมซอฟต์แวร์ในประเทศไทยและความต้องการแรงงานทางด้านซอฟต์แวร์พบว่า ในปีพ.ศ. 2565 เพิ่มขึ้นจากปีพ.ศ. 2563 มากกว่าร้อยละ 16 เปอร์เซ็นต์ และมีแนวโน้มเพิ่มมากขึ้น (รายงานสถิติข้อมูลของอุตสาหกรรมซอฟต์แวร์ พ.ศ. 2565)

แม้ว่าผู้สำเร็จการศึกษาทางด้านเทคโนโลยีจะมีจำนวนมาก แต่บุคลากรเหล่านี้ก็ยังขาดทักษะการเป็นบุคลากรซอฟต์แวร์ที่สอดคล้องกับความต้องการของตลาดแรงงาน ในทักษะด้านเทคนิค (Technical skill) เช่น ภาษาการเขียนโปรแกรม เฟรมเวิร์ก กระบวนการพัฒนาซอฟต์แวร์ และทักษะเชิงบุคคล (Soft skill) เช่น ทักษะการสื่อสาร ซึ่งในแต่ละองค์กรมีความต้องการหลากหลายและเปลี่ยนแปลงตลอดเวลา ด้วยเหตุนี้อาจจะทำให้หลายๆ มหาวิทยาลัยที่มีการผลิตบัณฑิตในสาขานี้ได้รับผลกระทบจากการผลิตบัณฑิตไม่สอดคล้องกับความต้องการของตลาดแรงงานในอุตสาหกรรมซอฟต์แวร์และความเปลี่ยนแปลงของความต้องการของผู้ใช้บัณฑิต

งานวิจัยนี้จึงต้องการเก็บรวบรวมข้อมูลทักษะในแต่ละตำแหน่งงานทางด้านซอฟต์แวร์ตามความต้องการขององค์กรซอฟต์แวร์เพื่อนำมาจำแนกทักษะที่เป็นที่ต้องการ ทั้งทักษะด้านเทคนิค (Technical skill) และทักษะเชิงบุคคล (Soft skill) ของแต่ละตำแหน่งงานทางด้านซอฟต์แวร์ เพื่อนำไปต่อยอดการสร้างตัวแบบสมรรถนะที่พึงประสงค์ของบัณฑิตเพื่อสนองความต้องการของอุตสาหกรรมซอฟต์แวร์ในประเทศไทย และนำเสนอผลการวิจัยที่ได้ต่อหลักสูตรที่ผลิตบัณฑิตในสาขาดังกล่าวสำหรับเป็นข้อมูลสำหรับพัฒนาบัณฑิตให้สอดคล้องกับความต้องการของตลาดแรงงานในอนาคต

1.2 วัตถุประสงค์งานวิจัย

1) เพื่อศึกษาข้อมูลทักษะที่ต้องการขององค์กรซอฟต์แวร์ที่ประกาศรับตำแหน่งงานด้านซอฟต์แวร์จากเว็บไซต์รับสมัครงานชั้นนำในประเทศไทย

2) เพื่อนำเสนอชุดของทักษะตามความต้องการในแต่ละตำแหน่งงานของวิศวกรซอฟต์แวร์ไทย

3) เพื่อสร้างตัวแบบสำหรับประเมินตำแหน่งงานที่เหมาะกับสมรรถนะในตำแหน่งงานที่กำหนด

1.3 ขอบเขตและกรอบแนวคิดการวิจัย

1) งานวิจัยนี้เก็บรวบรวมข้อมูลจากเว็บไซต์รับสมัครงานในประเทศไทยที่มีการอนุญาตให้เข้าถึงข้อมูลได้แบบสาธารณะ ได้แก่ Jobthai.com และ Linkedln.com

2) ผู้วิจัยเก็บรวบรวมรายละเอียดทักษะที่ต้องการจากคำอธิบาย (Job description) แต่ละตำแหน่งงาน โดยข้อมูลที่เข้าเงื่อนไขจะใช้เฉพาะภาษาอังกฤษเท่านั้น ข้อมูลที่เก็บรวบรวมในงานวิจัยนี้จะดำเนินการเฉพาะวันที่ประกาศรับสมัคร 3 ปีย้อนหลัง (พ.ศ 2565-2567)

3) ข้อมูลที่ใช้ในงานวิจัยนี้เก็บข้อมูลที่เกี่ยวข้องกับตำแหน่งงานที่ใช้ในงานวิจัย (Software Engineering Roles version 3.0, 4.0 : SWEBOK) [4-5] จำนวน 7 ตำแหน่งงาน ประกอบด้วย

1. วิศวกรความต้องการ (Requirements Engineer)
2. นักออกแบบซอฟต์แวร์ (Software Designer)
3. นักพัฒนาซอฟต์แวร์ (Developer)
4. นักทดสอบคุณภาพซอฟต์แวร์ (Software Tester)
5. วิศวกรซ่อมบำรุง (Maintenance Engineer)
6. ผู้จัดการโครงการซอฟต์แวร์ (Project Manager)
7. นักวิเคราะห์ระบบ (System Analyst)

4) คำที่ใช้ในการวิเคราะห์จะเป็นชุดของคำที่เกี่ยวข้องกับทักษะด้านเทคนิค (Hard skill) ซึ่งแบ่งเป็นกลุ่มคือ ทักษะด้านเทคนิค (Technical skill) ออกเป็น 6 กลุ่มทักษะย่อย ได้แก่ ภาษาการเขียนโปรแกรม (Programming languages) แพตฟอร์ม (Platform) การจัดเก็บและฐานข้อมูล (Database and Data Usage) เฟรมเวิร์ก (Framework) เครื่องมือสำหรับพัฒนา (Development tools และ เครื่องมือทดสอบ (Testing tools) และทักษะเชิงบุคคล (Soft skill) ผู้วิจัยใช้ชุดทักษะที่กำหนดไว้ใน SFIA เวอร์ชัน 8 (The global skills and competency framework for the digital world) และการทบทวนวรรณกรรมตามงานวิจัยของ [5-16] ดังนี้

ตารางที่ 1-1 กลุ่มคำของทักษะด้านบุคคล (Soft skill)

#	Soft skill		Soft skill	#	Soft skill
1.	Activeness	13.	Essence	25.	Research
2.	Analytical	14.	Estimate timeline	26.	Responsibility
3.	Collaborate	15.	Flexibility	27.	Result orientation
4.	Commitment	16.	innovation	28.	Self Management
5.	Communication	17.	Knowledge transfer	29.	Self-confidence
6.	Conceptual skills	18.	Leadership	30.	Self-esteem
7.	Creative	19.	Lift long learning	31.	Self-organization
8.	Critical thinking	20.	Motivation	32.	Situation awareness
9.	Curiosity	21.	Positive attitude	33.	social competence
10.	Customer orientation	22.	Pressure	34.	Teamwork
11.	Decision making	23.	Problem-solving	35.	Visionary
12.	Ethical/work ethic	24.	Quality control	36.	Willingness

5) สร้างตัวแบบ Classification โดยใช้ตัวแบบในกลุ่ม Machine learning วิธี Random Forest, Naïve Bayes Support, XGBoost และการวิเคราะห์ LSTM ซึ่งอยู่ในกลุ่มการเรียนรู้เชิงลึก (Deep Learning)

6) ตัววัดคุณภาพของโมเดลด้วย Accuracy, Precision, Recall และ F1-score

1.4 นิยามศัพท์เฉพาะ

1) สมรรถนะของวิศวกรซอฟต์แวร์ (Competency of software engineer) หมายถึง ความสามารถในการนำความรู้ (Knowleng) ทักษะเทคนิค (Technical skill) และทักษะด้านบุคคล (Soft skill) มาใช้ในสถานการณ์จริง เพื่อให้บรรลุผลสำเร็จของงานอย่างมีประสิทธิภาพ

2) ทักษะด้านเทคนิค (Technical skill) หมายถึง ความรู้ ความสามารถ หรือความชำนาญเฉพาะทาง ที่เกี่ยวข้องกับการใช้งานเครื่องมือ เทคโนโลยี ซอฟต์แวร์ หรือกระบวนการทางวิชาชีพ เพื่อให้สามารถปฏิบัติงานได้อย่างมีประสิทธิภาพ

3) ทักษะด้านบุคคล (Soft skill) หมายถึง ความสามารถเชิงพฤติกรรมและปฏิสัมพันธ์ของบุคคล ที่ช่วยให้ทำงานร่วมกับผู้อื่นได้อย่างมีประสิทธิภาพ เป็นทักษะที่ไม่ใช่เชิงเทคนิคแต่สำคัญต่อความสำเร็จในการทำงานและการใช้ชีวิต

4) การทำเหมืองข้อความ (Text mining) การสกัดองค์ความรู้ที่เป็นประโยชน์จากฐานข้อมูลขนาดใหญ่ โดยมีกระบวนการทำงาน 3 ขั้นตอน ได้แก่ การรวบรวมข้อมูล (Data collection) การประมวลผลข้อความ (Text processing) และการวิเคราะห์ข้อความ (Text analysis)

1.5 ประโยชน์ที่ได้รับจากงานวิจัย

1) สถานศึกษาสามารถนำสารสนเทศที่ได้ไปประกอบการประเมินจุดแข็งและจุดอ่อนของหลักสูตรในการพัฒนาผู้เรียนให้มีสมรรถนะตามความต้องการขององค์กรได้

2) ผู้บริหารสถานศึกษาสามารถนำแนวทางการส่งเสริมสมรรถนะไปประกอบการกำหนดนโยบายและการพัฒนาหลักสูตรทางวิศวกรรมซอฟต์แวร์ให้เหมาะสม

3) ผู้กำหนดนโยบายสามารถนำข้อมูลสมรรถนะของวิศวกรซอฟต์แวร์ในตำแหน่งต่างๆ ไปพัฒนาหลักสูตรทั้งระยะสั้นและระยะยาวให้สอดคล้องกับความต้องการของตลาดแรงงานไทยได้

บทที่ 2
ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 สมรรถนะ (Competence)

สมรรถนะ (Competence) หมายถึง ความสามารถในงานหรือเป็นคุณลักษณะที่อยู่ภายในบุคคลที่นำไปสู่ การปฏิบัติงานให้เกิดประสิทธิภาพ ประกอบด้วย ความรู้ ทักษะ และเจตคติ [17-19] ในขณะเดียวกัน Hodkinson and Issitt [20] ได้นำเสนอ สมรรถนะแบบองค์รวมเพื่อเพิ่มความสามารถในวิชาชีพในการบูรณาการความรู้ความเข้าใจ ค่านิยมและ ทักษะที่อยู่ในตัวผู้ที่เป็นผู้ปฏิบัติงานซึ่งสอดคล้องกับแนวคิดของ Cheetham and Chivers [21] ที่ว่าการพัฒนาสมรรถนะแบบองค์รวมจะประกอบด้วย 5 สมรรถนะที่สามารถเชื่อมโยงถึงกันได้ ดังนี้

1) สมรรถนะทางปัญญา (Cognitive Competences) เกี่ยวกับความรู้ทางแนวคิด ทฤษฎี

2) สมรรถนะในการทำงาน (Functional Competences) เกี่ยวกับทักษะที่บุคคลจะต้องทำ ได้ในสายอาชีพ

3) สมรรถนะส่วนบุคคล (Personal Competences) เกี่ยวกับความสามารถทางพฤติกรรม ส่วนบุคคลและรู้วิธีการปฏิบัติตนในการปฏิบัติงาน

4) สมรรถนะจริยธรรม (Ethical Competences) เกี่ยวกับการที่บุคคลประพฤติ ปฏิบัติตน ได้อย่างเหมาะสมกับค่านิยมทางวิชาชีพและกฎระเบียบเกี่ยวกับงาน

5) สมรรถนะการเรียนรู้ตลอดชีวิต (Meta Competences) เกี่ยวข้องกับความสามารถใน การรับมือกับความไม่แน่นอน สามารถเรียนรู้และเข้าใจตนเอง มีความยืดหยุ่นทางความคิด มีความคิด สร้างสรรค์ สามารถบริหารจัดการชีวิตได้

Le Deist and Winterton [22] ได้จัดประเภทสมรรถนะแบบองค์รวมเพื่อประโยชน์ในการทำความเข้าใจโดยผสมผสานความรู้ทักษะ และความสามารถทางสังคมที่จำเป็น สำหรับการประกอบอาชีพ โดยจัดออกเป็น 2 มิติ มิติแรกเป็นความสามารถในการทำงาน (Occupational) ประกอบด้วย สมรรถนะด้านความรู้ (Cognitive Competencies) และสมรรถนะ ด้านการทำงาน (Functional Competencies) ส่วนมิติที่ 2 เป็นความสามารถส่วนบุคคล (Personal Competencies) ประกอบด้วยสมรรถนะด้านการเรียนรู้ตลอดชีวิต (Meta Competencies) และ สมรรถนะด้านสังคม (Social Competencies) ดังตารางที่ 2-1

เพื่อให้ได้บัณฑิตที่มีคุณภาพและสมรรถนะในการแข่งขันทั้ง ตลาดแรงงานในประเทศและต่างประเทศได้ รวมไปจนถึงการดำเนินการพัฒนาทักษะที่จำเป็นต่อ การดำรงชีวิตในยุคดิจิทัล และสามารถพร้อมรับการเปลี่ยนแปลงในรูปแบบต่าง ๆ ได้ดังนั้น การพัฒนา สมรรถนะและความสามารถของนักศึกษาเพื่อที่จะออกไปเผชิญกับสถานการณ์ในโลกการทำงานจริงนั้น จึงไม่ได้เน้นเฉพาะการถ่ายทอดองค์ความรู้เท่านั้น แต่ต้องเสริมสร้างทักษะที่จำเป็นต่อการทำงาน เช่น ทักษะในศตวรรษที่ 21 ซึ่งประกอบด้วย 3Rs โดยผู้เรียนจะต้อง

อ่านออก เขียนได้ และสามารถคิดเลขเป็น และ 8Cs ยกตัวอย่าง เช่น ทักษะด้านการคิดอย่างมีวิจารณญาณ ทักษะในการแก้ปัญหา มีความคิด สร้างสรรค์สามารถทำงานเป็นทีมได้ที่สำคัญจะต้องมีทักษะทางด้านเทคโนโลยีสารสนเทศและการสื่อสาร ทั้งหมดนี้จะต้องควบคู่ไปกับการมีคุณธรรม จริยธรรม เมื่อนักศึกษามีความรู้ในสาขาวิชา รู้รอบในศาสตร์ ที่เกี่ยวข้องและมีทักษะต่าง ๆ ดังกล่าวข้างต้น จึงมีความพร้อมในขั้นต้นต่อการรับมือในสภาพแวดล้อมที่ แปรเปลี่ยนไปของการทำงานหรือองค์ความรู้ใหม่ ๆ ในอนาคต [23]

ตารางที่ 2-1 ประเภทของสมรรถนะ

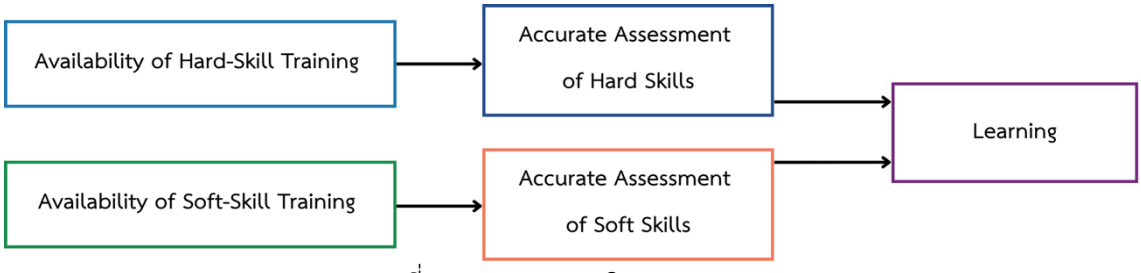
มิติ (Dimension)	ความสามารถในการทำงาน (Occupational)	ความสามารถส่วนบุคคล (Personal)
แนวคิด (Conceptual)	สมรรถนะด้านความรู้ (Cognitive Competencies)	สมรรถนะด้านการเรียนรู้ตลอดชีวิต (Meta Competencies)
การปฏิบัติงาน (Operation)	สมรรถนะด้านการทำงาน (Functional Competencies)	สมรรถนะด้านสังคม (Social Competemcies)

ที่มา: Le Deist & Winterton [22]

นอกจากนั้นการพัฒนานิสิตในสถาบันให้ประสบความสำเร็จต้องมีการบูรณาการระหว่างทักษะวิชาชีพ (Hard Skills) และทักษะคนและสังคม (Soft Skills) เพื่อให้นิสิตเป็นบัณฑิตที่มีความสมดุลทั้งการเป็นคนเก่งและเป็นคนดีอย่างเหมาะสม ซึ่งการพัฒนาให้นักศึกษามีศักยภาพทางด้าน มิติ (Dimension) ความสามารถในการทำงาน (Occupational) และความสามารถส่วนบุคคล (Personal) แนวคิด (Conceptual) สมรรถนะด้านความรู้ (Cognitive Competencies) และสมรรถนะด้านการเรียนรู้ตลอดชีวิต (Meta Competencies) ส่วนการปฏิบัติงาน (Operational) มีสมรรถนะด้านการทำงาน (Functional Competencies) และสมรรถนะด้านสังคม (Social Competencies)

Hard Skills จะเป็นการพัฒนาทักษะวิชาชีพหลักในวิชาชีพที่เกี่ยวข้องกับหน้าที่การงาน โดยตรงและที่จำเป็นในการปฏิบัติงาน [24-25] ในขณะที่การเสริมสร้างความสามารถทางด้านคนและสังคม หรือ Soft Skills จะเป็นทักษะหรือความชำนาญ ที่เกี่ยวข้องกับสังคมหรือความสัมพันธ์ระหว่างบุคคล ซึ่งจะช่วยให้การทำงานประสบความสำเร็จและ ก้าวหน้า เช่น ทักษะการสื่อสาร ทักษะการแก้ปัญหา ภาวะผู้นำ การบริหารจัดการ การทำงานเป็นทีม การควบคุมอารมณ์และความรับผิดชอบต่อสังคม ซึ่งบุคคลที่มี Soft Skills จะสามารถปฏิบัติงานได้ อย่างมีประสิทธิภาพสูง และทำให้ลดปัญหาสังคม ในเรื่องความสัมพันธ์ของเพื่อนร่วมงานในที่ทำงาน [26] ทั้ง Hard Skills และ Soft Skills มีความจำเป็นและมีความสำคัญสำหรับทุกอาชีพขึ้นอยู่กับ ลักษณะของงานหรืออาชีพที่จะเป็นตัวกำกับและบ่งบอกถึงน้ำหนักของ Hard Skills และ Soft Skills ว่าในลักษณะงานใดมีความต้องการ Hard Skills หรือ Soft Skills มากกว่ากัน โดยลักษณะงานในอาชีพต่าง ๆ ในปัจจุบันมีอาชีพจำนวนมากมายและเกิดขึ้นใหม่อีกจำนวนมาก การพัฒนาบุคคลให้ มีความพร้อมทั้ง Hard Skills และ Soft Skills ไปพร้อม ๆ กัน จึงเป็นสิ่งที่มีความจำเป็นอย่างมาก สำหรับคนรุ่นใหม่

ที่กำลังจะเข้าสู่การประกอบอาชีพในยุคดิจิทัล ดังนั้น การศึกษาจะต้องมีการเปลี่ยนแปลงให้มีลักษณะองค์รวม (Holistic) ที่จะต้องอาศัย อาจารย์และบุคลากรที่เข้าใจและมีความสามารถในการบูรณาการ Soft Skills ที่ส่งเสริม Hard Skills ทั้งใน และนอกชั้นเรียน ซึ่งสอดคล้องกับกรอบแนวคิดของ Zhang [25] ที่ได้วิจัยโดยการออกแบบและ กำหนดการประเมินเพื่อส่งเสริม Soft Skillsและ Hard Skills และการประเมินที่แม่นยำจะส่งเสริมและกำหนด รูปแบบการเรียนรู้ที่เกิดขึ้นได้ [27-28] ดังแผนภาพ 2.1



ภาพที่ 2-1 กรอบแนวคิดของ Zhang (2012)

จากการทบทวนแนวคิดเกี่ยวกับแนวคิดเกี่ยวกับ Hard Skills และ Soft Skills พบว่าสอดคล้องกับงานวิจัยของ Salman และคณะ [29] ที่ได้ศึกษาแนวคิดของ สมรรถนะ: การทบทวนวรรณกรรมอย่างเป็นระบบและการอภิปราย โดยนำเสนอกรอบแนวคิดของ สมรรถนะและแนวคิด Hard Skills และ Soft Skills ร่วมกัน ดังตาราง 2-2

ตารางที่ 2-2 กรอบแนวคิด Hard Skills และ Soft Skills

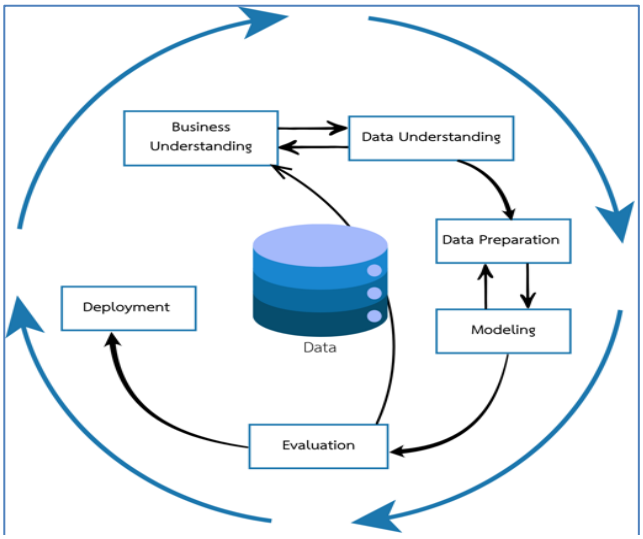
Hard skill	Soft skill
H1. สมรรถนะที่เกี่ยวข้องกับความรู้ (Knowledge Related Competence) H1.1 Cognitive Competence H1.2 Conceptual Competence	S1. สมรรถนะที่เกี่ยวข้องกับพฤติกรรม (Behavior-Related Competence) S1.1 Social/Behavior Competence S1.2 Operation Competence S1.3 Emotion Competence S1.4 Cross-Cultural Competence S1.5 Team Competence S1.6 Communication Competence
H2. สมรรถนะที่เกี่ยวข้องกับทักษะ (Skill Related Competence) H2.1 Functional Competence H2.2 Occupation Competence H2.3 Job Competence	S2. สมรรถนะที่เกี่ยวข้องกับการตระหนักรู้ในตนเอง (Self-Actualization-Related Competence) S2.1 Meta Competence S2.2 Ethicval Competence S2.3 Self Competence S2.4 Change Competence

จากแนวคิดดังกล่าวข้างต้น ผู้วิจัยจึงนำกรอบการจัดหมวดหมู่คุณลักษณะที่พึงประสงค์ในยุคดิจิทัลของ Salman และคณะ [29] และ Le Deist และ Winterton [30] มาเป็นขอบเขต โดยจัด กลุ่มสมรรถนะของวิศวกรรมซอฟต์แวร์ออกเป็น 2 มิติ โดยมิติแรกจำแนกตามทักษะการทำงาน (Employability Skills) ประกอบด้วย ทักษะทางวิชาชีพ (Hard Skill) และทักษะทางสังคม (Soft Skill)

2.2 การวิเคราะห์ข้อความ (Text Mining)

การวิเคราะห์ข้อความ (Text Analysis) เป็นกระบวนการที่คอมพิวเตอร์พยายามทำความเข้าใจกับภาษาธรรมชาติของมนุษย์ โดยได้นำองค์ความรู้จากหลากหลายสาขามาประยุกต์ใช้ เช่น การค้นหาลักษณะแฝงของข้อมูล (Knowledge Discovery), การทำเหมืองข้อมูล (Data Mining), และการประมวลผลภาษาธรรมชาติ (Natural Language Processing: NLP) เป็นต้น สำหรับการวิเคราะห์ข้อความถือเป็นกระบวนการในการค้นหาและสกัดความรู้จากฐานข้อมูลขนาดใหญ่ (Large Textual Information) เพื่อให้ได้สารสนเทศที่เป็นประโยชน์ (Useful Textual Information) โดยข้อมูลที่นำมาวิเคราะห์จะอยู่ในรูปแบบของข้อความ (Text Data Sets) หรือภาษาธรรมชาติ (Natural Language) จึงเรียกกระบวนการนี้ว่า “การวิเคราะห์เหมืองข้อความ” (Text Mining) ซึ่งได้นำองค์ความรู้ในด้านการค้นหาลักษณะแฝงของข้อมูลมาประยุกต์ใช้ในการวิเคราะห์ การค้นหาลักษณะแฝงของข้อมูล (Knowledge Discovery in Data: KDD) หรือการค้นหาความรู้จากฐานข้อมูลขนาดใหญ่ (Large Database) เป็นกระบวนการที่มุ่งค้นหารูปแบบ โครงสร้างความสัมพันธ์ หรือการเปลี่ยนแปลงที่แฝงอยู่ในข้อมูล โดยข้อมูลที่นำมาวิเคราะห์อาจอยู่ในรูปของฐานข้อมูล ข้อความ หรือแม้แต่รูปภาพ การวิเคราะห์เหมืองข้อมูลถือเป็นกระบวนการหลักของการค้นหาลักษณะแฝงในข้อมูล (Knowledge Discovery) ซึ่งบางครั้งการวิเคราะห์เหมืองข้อมูล หมายถึงกระบวนการค้นหาลักษณะแฝงของข้อมูลเลยก็ได้ โดยหน่วยงาน Cross Industry Standard Process for Data Mining (Crisp-DM) ได้นำเสนอกระบวนการวิเคราะห์ข้อมูลประกอบด้วยขั้นตอนต่อไปนี้

1) การทำความเข้าใจกับธุรกิจและระบุปัญหาของงาน (Business Understanding)
2) การรวบรวมข้อมูลและพิจารณาความถูกต้อง เหมาะสมของข้อมูล (Data Understanding)
3) การเตรียมข้อมูล (Data Preparation)
4) การสร้างแบบจำลองการวิเคราะห์ข้อมูล (Modeling)
5) การประเมินหรือวัดประสิทธิภาพของแบบจำลอง (Evaluation)
6) การนำผลลัพธ์จากการวิเคราะห์ไปใช้งานจริง (Deployment) ดังภาพที่ 2-2



ภาพที่ 2-2 กระบวนการค้นหาลักษณะแฝงของข้อมูล (Crisp DM)

แหล่งที่มา: IBM Software Business Analytics [31]

ลักษณะของข้อมูลที่นำมาวิเคราะห์เพื่อค้นหาลักษณะแฝง สามารถแบ่งออกเป็น 2 ประเภท คือ1) ข้อมูลที่เป็นโครงสร้าง (Structured Data) การประมวลผลข้อมูลที่เป็นโครงสร้าง เรียกว่าการวิเคราะห์เหมืองข้อมูล (Data Mining) และ 2) ข้อมูลที่ไม่เป็นโครงสร้างหรือไม่มีโครงสร้างที่แน่นอน (Unstructured or Implicit Structured Data) โดยส่วนใหญ่มักอยู่ในรูปแบบของข้อความหรือภาษาธรรมชาติ เรียกว่าการวิเคราะห์เหมืองข้อความ หรือกระบวนการค้นหาลักษณะแฝงข้อความ (Knowledge Discovery from Text: KDT)

สำหรับการวิเคราะห์ข้อความมีกระบวนการที่แตกต่างจากการวิเคราะห์เหมืองข้อมูลเล็กน้อย เนื่องจากข้อความ มีลักษณะข้อมูลแบบไม่เป็นโครงสร้างหรือไม่มีโครงสร้างที่แน่นอน จึงจำเป็นต้องมี กระบวนการแปลงรูปแบบของข้อความที่ไม่มีโครงสร้างให้กลายเป็นข้อมูลที่มีโครงสร้างก่อน ซึ่งเรียกกระบวนการนี้ว่า "การเตรียมข้อมูล" (Data Preparation) ขั้นตอนของการเตรียมข้อมูลในกระบวนการวิเคราะห์ข้อความถือเป็นขั้นตอนที่สำคัญและใช้เวลานาน ซึ่งแตกต่างจากการวิเคราะห์เหมืองข้อมูลแบบทั่วไป เพื่อให้ได้ข้อมูลที่มีโครงสร้างที่เหมาะสมสำหรับขั้นตอนวิธี (Algorithm) ที่จะนำมาใช้ในการวิเคราะห์ กระบวนการเตรียมข้อมูลเกี่ยวข้องกับเทคนิคการประมวลผลภาษาธรรมชาติ ที่นำมาช่วยในการเตรียมข้อความให้คอมพิวเตอร์สามารถทำความเข้าใจกับคำและประโยคในภาษาธรรมชาติได้มากขึ้น และผลลัพธ์ที่ได้จากการค้นหาลักษณะแฝงของข้อมูลคือองค์ความรู้ (Knowledge) หรือสารสนเทศที่เป็นประโยชน์ ดังนั้นกระบวนการวิเคราะห์เหมือง จึงประกอบด้วยการนำทฤษฎีการประมวลผลภาษาภาษาธรรมชาติ และการวิเคราะห์เหมืองข้อมูลมาประยุกต์ใช้ร่วมกันเป็นทฤษฎีหลักในการวิเคราะห์เหมืองข้อความ ซึ่งมีขั้นตอนวิธี ดังนี้

การเตรียมข้อมูลในการค้นหาลักษณะแฝงของข้อมูล สามารถแบ่งออกเป็น 3 กระบวนการหลัก ตามแนวทางของ IBM Software Business Analytics [32] ได้แก่ การคัดเลือกข้อมูล (Data Selection)

การกลั่นกรองข้อมูล (Data Cleaning) การสร้างข้อมูลที่เหมาะสม (Data Constructing) การรวมข้อมูล (Data Integration) และสุดท้ายคือการแปลงรูปข้อมูล (Data Transformation) ให้เป็นข้อมูลที่มีโครงสร้างที่เหมาะสม

สำหรับการวิเคราะห์เหมืองข้อความในงานวิจัยนี้ได้จำแนกกระบวนการเตรียมข้อมูลออกเป็น 3 กระบวนการหลัก ได้แก่

1) การเลือกคุณลักษณะ (Feature selection)

2) การกลั่นกรองข้อความ (Text cleaning)

3) การแทนข้อความ (Text representation)

กระบวนการเหล่านี้ได้อ้างอิงตามหลักการเตรียมข้อมูลของเทคนิคการวิเคราะห์เหมืองข้อมูล แต่สำหรับการวิเคราะห์เหมืองข้อความต้องอาศัยทฤษฎีการประมวลผลภาษาธรรมชาติในการวิเคราะห์ร่วมด้วย การสร้างแบบจำลองการวิเคราะห์ข้อมูล เป็นกระบวนการสร้างแบบจำลองขึ้นจากข้อมูลที่มีอยู่ เพื่อใช้ในการอธิบายรูปแบบของข้อมูลหรือทำนายรูปแบบของข้อมูลที่ยังไม่เกิดขึ้น โดยแบ่งออกเป็น 2 เทคนิคหลัก คือ

1) กระบวนการเรียนรู้แบบมีผู้สอน (Supervised Learning) เช่น การจำแนกข้อมูล (Classification) สำหรับการวิเคราะห์ข้อความ นิยมเรียกกระบวนการนี้ว่า การจำแนกข้อความ (Text Classification หรือ Text Categorization)

2) กระบวนการเรียนรู้แบบไม่มีผู้สอน (Unsupervised Learning) เช่น การจัดกลุ่มข้อมูล (Clustering)

การประเมินประสิทธิภาพของแบบจำลองการวิเคราะห์ข้อมูล ใช้เพื่อเป็นเครื่องมือวัดความน่าเชื่อถือของแบบจำลอง

จากกระบวนการวิเคราะห์เหมืองข้อความข้างต้น เมื่อนำมาประยุกต์ใช้กับงานวิจัยเกี่ยวกับการวิเคราะห์เหมืองข้อเสนอแนะ จำเป็นต้องอาศัยความรู้จาก 4 ทฤษฎีหลักคือ 1) การแทนข้อความ 2) การประมวลผลภาษาธรรมชาติ 3) การจำแนกประเภทข้อความ และ 4) การประเมินประสิทธิภาพของการจำแนกประเภทข้อความ มีรายละเอียดดังนี้

1) การแทนที่ข้อความ (Text Representation)

การวิเคราะห์ข้อความที่มีโครงสร้างไม่แน่นอน จำเป็นต้องแปลงให้อยู่ในรูปแบบที่มีโครงสร้าง ก่อนเพื่อให้คอมพิวเตอร์สามารถนำไปประมวลผลได้ การแทนข้อความในรูปแบบเวกเตอร์สเปซโมเดล (Vector Space Model: VSM) เป็นวิธีการหนึ่งในการแทนข้อความให้มีโครงสร้างแบบฟีเจอร์เวกเตอร์ (Feature Vector) สามารถเลือกคุณลักษณะของข้อความมาแทนได้หลายวิธี เช่น คำ วลี หรือหน้าที่ของคำ เป็นต้น โดยปกติแล้วการวิเคราะห์ข้อความจะอาศัยการวิเคราะห์จากคำเป็นหลัก เรียกว่าการแทนที่ข้อความด้วยถุงคำ (Bag-of-word) เป็นการแทนคำทุกคำในเอกสารด้วยเวกเตอร์ที่ประกอบด้วยสมาชิกคือ ค่าที่เป็นตัวแทนของคำภายในข้อความนั้น มีวิธีที่ใช้คำนวณค่าเพื่อกำหนดให้เป็นตัวแทนข้อความ เช่น

- การแทนคำด้วยค่าการเกิดขึ้นหรือไม่เกิดขึ้นของคำ (Binary weighting) ดังสมการ (1)
- การแทนด้วยค่าความถี่ของคำ (Term Frequency: TF)

- ค่าความถี่คำ-ค่าส่วนกลับความถี่เอกสารที่เกิดคำ (Term Frequency – Inverse Document Frequency: TF-IDF) ซึ่งวิธีการแทนค่าข้อความด้วยค่า TF-IDF เป็นวิธีที่ได้รับความนิยมมากที่สุด เนื่องจากเป็นวิธีการคำนวณที่ง่ายและมีประสิทธิภาพสูง

$$binary\ weighting = \begin{cases} 1, for\ term\ present\ in\ the\ document \\ 0, otherwise. \end{cases} \quad (1)$$

1.1) Term Frequency – Inverse Document Frequency: TF-IDF

แนวความคิดของการแทนข้อความด้วยค่า TF-IDF เกิดจากแนวความคิดว่าการแทนข้อความด้วยค่าความถี่ของคำเพียงอย่างเดียว ไม่สามารถจำแนกเอกสาร (หรือข้อความ) ได้ดีพอ เนื่องจากค่าความถี่ของคำสูง หมายถึงคำนั้นมีโอกาสเกิดขึ้นในหลายเอกสารพร้อมกัน จึงไม่มีประโยชน์ต่อการจำแนกเอกสาร ดังนั้น Salton and Buckley (1988) [33] จึงนำเสนอวิธีการแทนค่าเอกสารด้วยค่า TF-IDF คือคำนึงถึงความถี่ของคำในเอกสารด้วย โดยน้ำหนักของคำที่ได้ คือ W_d เกิดจากการคูณค่า TF ด้วย IDF ดังสมการ (2)

$$W_d = f_{w,d} * \log\left(\frac{|D|}{f_{w,d}}\right) \quad (2)$$

เมื่อ $f_{w,d}$ หมายถึง ความถี่ของคำ (Term Frequency: TF) เป็นค่าที่เกิดจากการคำนวณค่าความถี่ของคำ (w) ที่พบภายในเอกสาร (d) และค่า Logarithmic scale ของสัมประสิทธิ์ของจำนวนเอกสารทั้งหมดที่นำมาวิเคราะห์ (D) หารด้วยค่าความถี่เอกสารที่พบคำ (w) ซึ่งหมายถึงค่าส่วนกลับของเอกสาร (Inverse Document Frequency: IDF) หรือค่าส่วนกลับความถี่ของทุกเอกสาร (D) ที่ปรากฏคำ (w)

สำหรับการวิเคราะห์ข้อความในระดับประโยค จะแทนเอกสารด้วยประโยค ดังนั้นค่า TF หมายถึงค่าที่เกิดจากการคำนวณค่าความถี่ของคำ (w) ที่พบในประโยค (s) และค่า IDF หมายถึงค่า Logarithmic scale ของสัมประสิทธิ์ของจำนวนประโยคทั้งหมดที่นำมาวิเคราะห์ (D) หารด้วยค่าความถี่ของเอกสารที่พบคำ (w) Ramos, J. (1999).[34]

อย่างไรก็ตาม การแทนข้อความด้วยวิธีการคำนวณค่าจากคำทั้งหมดอาจไม่เหมาะสมนัก เนื่องจากมีคำบางคำที่ไม่มีประโยชน์ต่อการจำแนกข้อความ จึงควรเลือกคุณลักษณะของข้อความที่สามารถใช้เป็นตัวแทนเอกสารหรือข้อความที่ดี และตัดคำที่ไม่สามารถใช้เป็นตัวแทนเอกสารหรือข้อความที่ดีได้ออก การตัดคำไม่มีนัยสำคัญและการทำรากศัพท์ ถือเป็นวิธีการเลือกคุณลักษณะเบื้องต้นที่ดีและได้รับความนิยม เพื่อช่วยเพิ่มประสิทธิภาพการจำแนกข้อความ นอกจากนั้นแล้วยังสามารถลดขนาดของเนื้อที่เก็บข้อมูลได้มากถึง 30-50%

การตัดคำที่ไม่มีนัยสำคัญ (Stop Word Removal) หมายถึงคำที่เกิดขึ้นบ่อยในหลายเอกสารและไม่เป็นประโยชน์ต่อการจำแนกประเภทเอกสาร หากตัดคำเหล่านั้นออกจะไม่ทำให้ความหมายของประโยคเปลี่ยนไป เช่น คำเชื่อม (Conjunction) คำบุพบท (Preposition) คำหยุด (Ending Words) เป็นต้น

การทำรากศัพท์ (Word stemming) หมายถึงคำที่มีรากศัพท์คำเดียวกันแต่มีการแปลงรูปไป เช่น “run”, “ran” และ “runs” มีรากศัพท์คำเดียวกันคือ “run” แต่สำหรับภาษาไทยไม่มีคำในลักษณะดังกล่าว

มีเพียงคำที่มีความหมายคล้ายคลึงกันแต่ใช้คำต่างกัน (Synonym Word) เช่น คำว่า “กิน” กับ “รับประทาน” เป็นต้น

อย่างไรก็ตาม กระบวนการแทนข้อความด้วยวิธี Vector Space Model ที่แทนข้อความด้วยคำเพียงคำเดียว ไม่ได้วิเคราะห์จากกลุ่มคำหรือลำดับของคำ จึงมีงานวิจัยที่ได้นำเสนอกระบวนการในการเลือกคุณลักษณะที่พิจารณาการเกิดขึ้นร่วมกันของคำ ได้แก่ การเลือกคุณลักษณะสำคัญด้วยค่าการเพิ่มของข้อมูล (Information Gain: IG) ค่าข้อมูลร่วม (Mutual Information: MI) ค่าสถิติ Chi-square ซึ่งในงานวิจัยของ Jan and Yiming Yang (1997) [35] ได้นำแต่ละวิธีมาเปรียบเทียบประสิทธิภาพการจำแนกเอกสาร ซึ่งได้แก่ การเพิ่มของข้อมูล, ค่าข้อมูลร่วม, ค่าสถิติ Chi-square และค่าความถี่ของเอกสารที่เกิดคำ (Document Frequency: DF) พบว่า IG และ Chi-square ได้ผลลัพธ์การจำแนกเอกสารดีที่สุด ส่วน DF มีวิธีการคำนวณที่ง่ายกว่าวิธีอื่น และมีประสิทธิภาพการจำแนกที่ใกล้เคียงกับวิธี IG และ Chi-square

นอกจากนี้ ในงานวิจัยของ Maria and Osmar (2002) [36] และ Minqing Hu and Bing Liu (2004b) [37] ได้นำเสนอกระบวนการเลือกคุณลักษณะสำคัญด้วยเทคนิคการหากฎความสัมพันธ์ (Association Rules Mining) เพื่อวิเคราะห์หาความสัมพันธ์ระหว่างคำหรือฟีเจอร์ (Feature) ตั้งแต่ 2 คำขึ้นไปที่เกิดขึ้นร่วมกันบ่อย

1.2) กฎความสัมพันธ์ของข้อมูล (Association Rules Mining)

การสร้างกฎความสัมพันธ์ของข้อมูลเป็นกระบวนการหนึ่งในการทำเหมืองข้อมูล เพื่อหาความสัมพันธ์ซึ่งกันและกันของข้อมูลตั้งแต่ 2 ชุดขึ้นไปภายในกลุ่มข้อมูลขนาดใหญ่ ซึ่งถูกนำเสนอโดย Rakesh Agrawal and Ramakrishnan Srikant (1994) ตัวอย่างหนึ่งของกฎความสัมพันธ์ที่นิยมใช้คือ การวิเคราะห์การซื้อสินค้าของลูกค้า (Market-basket analysis) ซึ่งหมายถึงการวิเคราะห์หาความสัมพันธ์ของสินค้าที่มีแนวโน้มในการซื้อร่วมกันบ่อยภายในรายการเดียวกัน กระบวนการสร้างกฎความสัมพันธ์ของข้อมูล แบ่งเป็น 2 กระบวนการ คือ

1. การหาความสัมพันธ์ของข้อมูลที่เกิดขึ้นร่วมกัน (Frequent Itemset) และวัดผลความสัมพันธ์ระหว่างข้อมูลด้วย ค่าสนับสนุน (Support) ความสัมพันธ์ของชุดข้อมูลใดที่มีค่าสนับสนุนมากกว่าค่าสนับสนุนขั้นต่ำที่กำหนดไว้ (Minimum Support) จะเรียกความสัมพันธ์ของชุดข้อมูลนั้นว่า “Frequent Patterns” หรือ “Frequent Itemset” และกระบวนการถัดไปคือ

2. การสร้างกฎความสัมพันธ์ (Association Rules) เป็นกระบวนการนำชุดข้อมูลที่มีความสัมพันธ์กันภายใต้ค่าสนับสนุนขั้นต่ำที่กำหนด มาสร้างเป็นกฎความสัมพันธ์ (Association Rule)ความสัมพันธ์ของชุดข้อมูลใดที่มี ค่าความเชื่อมั่น มากกว่าค่าความเชื่อมั่นขั้นต่ำ (Minimum Confidence) ที่กำหนดไว้ จะเรียกความสัมพันธ์ของชุดข้อมูลนั้นว่า “Association Rule” ตามภาพที่ 2-3

ภาพที่ 2-3 ขั้นตอนการสร้างกฎความสัมพันธ์

ค่าสนับสนุน หมายถึง เปอร์เซ็นต์ของจำนวนข้อมูลที่มีสมาชิกสอดคล้องตามกฎ ต่อจำนวนข้อมูลทั้งหมด ตัวอย่างเช่น ชุดข้อมูล A กับ B จะถูกกำหนดเป็นกฎความสัมพันธ์ A ⇒ B ก็ต่อเมื่อมีค่าสนับสนุนมากกว่าค่าเปอร์เซ็นต์ต่ำสุดที่กำหนดไว้ สมการการคำนวณค่าสนับสนุนดังสมการที่ (3)

$$support(AB) = \frac{transactions\ contain\ AB}{total\ transactions} \quad (3)$$

ค่าความเชื่อมั่น หมายถึง เปอร์เซ็นต์ของจำนวนข้อมูลที่สอดคล้องตามกฎต่อจำนวนข้อมูลทั้งหมดที่มีสมาชิกตามกฎฝั่งซ้ายมือ โดยที่กฎความสัมพันธ์ A ⇒ B ต้องมีค่าความเชื่อมั่นมากกว่าค่าเปอร์เซ็นต์ต่ำสุดที่กำหนดไว้ดังสมการที่ (4)

$$confidence(A \rightarrow B) = \frac{support(AB)}{support(A)} \quad (4)$$

นอกจากการแทนข้อความด้วยค่าความถี่ของคำคำเดียว หรือกลุ่มคำแล้ว ยังสามารถใช้หน้าที่ของคำเป็นตัวแทนข้อความได้เช่นเดียวกัน สำหรับงานวิจัยนี้ได้นำเสนอวิธีการแทนข้อความด้วยคำ หน้าที่ของคำและกลุ่มคำ ซึ่งการแทนข้อความด้วยกลุ่มคำนั้น ได้ใช้เทคนิคกฎความสัมพันธ์ของข้อมูลมาหาความสัมพันธ์ของคำที่เกิดขึ้นร่วมกันบ่อย และคำนวณค่าตัวแทนข้อความด้วยค่า TF-IDF

2) การประมวลผลภาษาธรรมชาติ (Natural Language Process)

ภาษาเป็นเครื่องมือที่ในการสื่อสารทั้งกับมนุษย์ด้วยกันเองหรือแม้กระทั่งสื่อสารกับคอมพิวเตอร์ แต่ภาษาที่ใช้มีรูปแบบที่แตกต่างกันไป ภาษาที่มนุษย์ใช้สื่อสารกับคอมพิวเตอร์ เป็นภาษาที่มีโครงสร้างแน่นอน คอมพิวเตอร์สามารถนำไปประมวลผลได้ทันที แต่สำหรับภาษาที่มนุษย์ใช้สื่อสารกันเองเป็นภาษาที่มีโครงสร้างหรือรูปแบบเฉพาะตัว และเป็นไปตามธรรมชาติของการเรียนรู้ในสมองมนุษย์แต่ละคนซึ่งมีลักษณะที่แตกต่างกันไป เรียกว่า “ภาษาธรรมชาติ” เป็นภาษาไม่มีโครงสร้างที่แน่นอน การที่คอมพิวเตอร์พยายามทำความเข้าใจกับภาษาธรรมชาติที่มนุษย์ใช้สื่อสารกันนั้น สามารถทำได้ด้วยวิธีการแทนความรู้ การสร้างกฎเกณฑ์ และการประเมินค่าเพื่อหาความหมายของภาษา

การที่คอมพิวเตอร์จะเข้าใจภาษาธรรมชาติได้ดีเพียงไรนั้นขึ้นอยู่กับ 2 แนวทาง หนึ่งคือพัฒนาการทางด้านปัญญาประดิษฐ์ซึ่งเป็นวิธีการแทนความรู้ (Knowledge Representation) และอีกแนวทางหนึ่งคือการศึกษาและเข้าใจโครงสร้างทางภาษาศาสตร์แบบมีโครงสร้าง ซึ่งทั้งสองกระบวนการดังกล่าวถูกเรียกว่าการประมวลผลภาษาธรรมชาติ โดยระบบประมวลผลภาษาธรรมชาติจะรับข้อมูลอินพุทเป็นข้อความและแทนค่าข้อความด้วยแนวทางการวิเคราะห์ต่าง ๆ เช่น ความรู้ หรือโครงสร้าง เป็นต้นการประมวลผลภาษาธรรมชาติได้แบ่งระดับขั้นการวิเคราะห์ ดังนี้

1. การวิเคราะห์ระดับวจีวิภาค (Morphological Analysis) เป็นการวิเคราะห์ระดับคำ

2. การวิเคราะห์ระดับวากยสัมพันธ์ (Syntactic Analysis) เป็นการวิเคราะห์คำตามหน้าที่ของคำ (Part-of-Speech) เพื่อเป็นข้อมูลพื้นฐานในการตรวจสอบโครงสร้างทางไวยากรณ์เกี่ยวกับการวางตำแหน่งของคำ กลุ่มคำประเภทต่าง ๆ ที่รวมกันเป็นประโยค

3. การวิเคราะห์ระดับความหมาย (Semantic Analysis) เป็นการวิเคราะห์เพื่อให้ทราบความหมายของแต่ละคำในประโยค

4. การวิเคราะห์ระดับวจนิพจน์ (Discourse Integration) เป็นการพิจารณาความหมายของประโยคโดยดูจากประโยคข้างเคียงร่วมด้วย

5. การวิเคราะห์ระดับปฏิบัติ (Pragmatic Analysis) เป็นการแปลความหมายของประโยคถึงสิ่งที่ผู้พูดต้องการสื่อความหมายถึง

กระบวนการวิเคราะห์ภาษาจะเริ่มต้นที่ระดับต่ำสุดก่อน คือ การวิเคราะห์ระดับวจีวิภาคหรือคำไปจนถึงระดับวากยสัมพันธ์ ที่สามารถอธิบายได้ด้วยโครงสร้างของภาษาที่ประกอบด้วยหน่วยต่าง ๆ ดังต่อไปนี้

1. คำ (Word) คือ หน่วยหนึ่งที่เปล่งเป็นเสียงออกมาจะเป็นอิสระหรือไม่ก็ได้

2. หน่วยคำ (Morpheme) คือ ส่วนประกอบที่มีนัยสำคัญที่แท้จริงทางภาษา คือ คำและกลุ่มคำ หรือหมายถึงหน่วยคำที่เล็กที่สุดที่มีความหมาย ความแตกต่างระหว่างหน่วยคำกับคำ คือ หน่วยคำอาจเกิดขึ้นอิสระหรือไม่อิสระก็ได้ แต่คำต้องเป็นหน่วยอิสระเสมอ เช่น นักเรียน “นัก” เป็นหน่วยคำ และ “นักเรียน” ถือเป็นคำ

3. วลี (Phrase) คือ กลุ่มของคำหรือคำ ๆ เดียวก็ได้ ที่เป็นส่วนประกอบของประโยค (วลีเป็นส่วนประกอบของประโยค แต่คำไม่ใช่ส่วนประกอบของประโยค) ดังนั้นไม่ว่าจะเป็นคำ ๆ เดียวหรือกลุ่มที่ประกอบด้วยหลาย ๆ คำ เมื่อเป็นส่วนประกอบของประโยค เราเรียกว่า วลี

4. นามวลี (Noun Phrase) คือ วลีที่ทำหน้าที่เป็นส่วนของประโยคที่เป็นหน่วยประธาน หน่วยกรรม

5. กริยาวลี (Verb phrase) คือ วลีที่ทำหน้าที่เป็นภาคแสดงของประโยค

6. ประโยค (Sentence) คือ คำหลายคำเรียงกันในการพูดหรือเขียน เพื่อเป็นการแสดงความคิด หนึ่งความคิดอย่างสมบูรณ์ทางไวยากรณ์ โดยปกติมักประกอบไปด้วยประธานและภาคแสดง

7. ไวยากรณ์ (Grammar) คือ ภาษาที่มีกฎเกณฑ์ และเป็นไปตามระเบียบวิธีการประกอบรูปคำให้เป็นประโยค

2.1) กระบวนการตัดคำ

ในการวิจัยด้านการตัดคำได้แบ่งการตัดคำออกเป็น 3 เทคนิคหลัก ได้แก่ 1) การตัดคำโดยใช้กฎ (Rule based approach) เป็นวิธีการพิจารณาการตัดคำที่ง่ายที่สุดและทำงานได้เร็วที่สุด 2) การตัดคำโดยใช้ฐานความรู้จากพจนานุกรม (Dictionary approach) 3) การตัดคำโดยใช้ค่าสถิติจากคลังเอกสาร (Corpus base approach) วิธีการนี้นำเอาสถิติการเกิดคำและหน้าที่ของคำเข้ามาช่วยในการคำนวณหาความน่าจะเป็นเพื่อเลือกแบบที่มีโอกาสการเกิดมากที่สุด วิธีการนี้มีความถูกต้องมากกว่า 2 วิธีแรก แต่มีข้อจำกัดคือ ต้องมี

ฐานข้อมูลที่ขนาดใหญ่มากพอ และมีการตัดคำไว้อย่างถูกต้องจึงจะทำให้ได้สถิติข้อมูลที่มีความน่าเชื่อถือ สำหรับงานวิจัยนี้ได้เลือกใช้การตัดคำแบบใช้ฐานความรู้จากพจนานุกรม แบบเลือกตัดคำจากคำที่พบในพจนานุกรม ซึ่งมีความเร็วในการประมวลผลและมีความถูกต้องระดับหนึ่ง แต่ปัญหาหลักของการประมวลผลธรรมชาติคือ ความกำกวมของภาษา เนื่องจากมีกฎเกณฑ์และข้อยกเว้นมากมาย คำแต่ละคำมีความหมายได้แตกต่างกัน ผู้วิจัยจึงตรวจสอบและแก้ไขผลลัพธ์ที่ได้จากกระบวนการตัดคำให้มีความถูกต้องก่อนนำเข้าสู่กระบวนการวิเคราะห์

2.2) กระบวนการกำกับคำตามหน้าที่ของคำ

การกำกับคำตามหน้าที่ของคำและการวิเคราะห์ในระดับวากยสัมพันธ์จะช่วยให้เข้าใจประโยคได้ถูกต้องยิ่งขึ้น ซึ่งการกำกับคำตามหน้าที่ของคำเป็นวิธีการบ่งบอกว่าคำแต่ละคำในประโยคเป็นคำชนิดใด จึงทำให้การวิเคราะห์ไวยากรณ์โครงสร้างประโยคและการแปลความถูกต้องยิ่งขึ้นหน้าที่หลักของคำแบ่งได้ดังนี้ คำนาม (Noun) คำสรรพนาม (Pronoun) คำกริยา (Verb) คำกริยาช่วย (Auxiliary Verb) คำวิเศษ (Adjective) คำบุพบท (Preposition) คำสันธาน (Conjunction) คำนามชี้เฉพาะ (Demonstrative Noun) คำนามบอกลักษณะ (Attributive Noun) คำปฏิเสธ (Negative word) และคำหยุด (Stop word)

3) การจำแนกประเภทข้อความ

การจำแนกประเภทข้อความ (Text Categorization หรือ Text Classification) มีเป้าหมายเพื่อสร้างแบบจำลองจากชุดข้อมูลเรียนรู้ที่รู้ผลเฉลยแล้ว (Train set) และนำแบบจำลองดังกล่าวไปจัดกลุ่มให้กับชุดข้อมูลทดสอบหรือชุดข้อมูลที่ยังไม่รู้ผลเฉลย (Test set) ให้อยู่ในกลุ่มที่กำหนดไว้ ซึ่งวิธีการจัดกลุ่มข้อความจะอาศัยการวิเคราะห์จากคำภายในข้อความเป็นหลัก โดยแบบจำลองการเลือกกลุ่มที่ดีที่สุดเกิดขึ้นจากการเรียนรู้จากชุดข้อมูลเรียนรู้ที่มีการจัดกลุ่มไว้แล้วโดยผู้เชี่ยวชาญ เรียกว่าการเรียนรู้แบบมีผลเฉลย วิธีการจำแนกข้อความแบ่งออกเป็น 2 วิธีหลักได้แก่

3.1) วิธีวิศวกรรมองค์ความรู้ (Knowledge Engineering Approach)

เป็นวิธีการสร้างกฎการจำแนกข้อความแบบ ถ้า – แล้ว ด้วยผู้เชี่ยวชาญ โดยวิธีการระบุคุณลักษณะสำคัญของข้อความและกำหนดกลุ่มที่เหมาะสมให้กับข้อความ ข้อดีคือ ได้กฎที่มีความถูกต้องแม่นยำสูง สามารถแก้ไขและจัดการง่ายหากมีจำนวนกฎน้อย ข้อเสียคือ อาจเกิดความขัดแย้งของกฎหากมีผู้เชี่ยวชาญสร้างกฎมากกว่าหนึ่งคน และต้องสร้างกฎใหม่ทุกครั้งที่เปลี่ยนโดเมนของชุดข้อมูล ทำให้ไม่เหมาะกับการจำแนกข้อความที่มีปริมาณมาก

3.2) วิธีการเรียนรู้ของเครื่อง (Machine Learning Approach)

เป็นวิธีการที่ใช้ในการเรียนรู้จากชุดข้อมูลเรียนรู้ เพื่อนำมาสร้างเป็นแบบจำลองแบบอัตโนมัติในการจำแนกข้อความ โดยข้อมูลที่นำมาเรียนรู้นั้นจะต้องมีการกำหนดผลเฉลยไว้แล้ว เพื่อให้คอมพิวเตอร์สามารถเรียนรู้รูปแบบของข้อมูล และสร้างแบบจำลองเพื่อไว้ใช้ทำนายหรือจัดกลุ่มของชุดข้อมูลทดสอบได้ ยิ่งมีปริมาณชุดข้อมูลเรียนรู้มากจะยิ่งทำให้แบบจำลองการจำแนกข้อความที่มีความถูกต้องสูง แต่อาจต้องใช้

ระยะเวลาในการสร้างแบบจำลองมากตามไปด้วย โมเดลสำหรับการจำแนกประเภทข้อความ เช่น Decision tree, Naïve baye และ Support vector machine เป็นต้น

ในปัจจุบันการเรียนรู้ของเครื่องได้มีการพัฒนาไปเป็นการเรียนรู้เชิงลึก (Deep Learning)ซึ่งการเรียนรู้เชิงลึกมีจุดเด่นในการเรียนรู้คุณลักษณะต่าง ๆ ได้อย่างหลายหลาย และเรียนรู้ได้จำนวนมาก (Goodfellow et al., 2016)โดยสามารถทำการสร้างแบบจำลองในการจำแนกประเภทของข้อความได้อย่างแม่นยำมากยิ่งขึ้น ซึ่งได้มีการนำมาประยุกต์ใช้ทางด้าน Natural Language Processing ในด้านของการสร้างแบบจำลองสำหรับแปลภาษา (Machine Translation) โดยสามารถแปลได้อย่างถูกต้องและแม่นยำ สืบเนื่องด้วยความสามารถทางเทคโนโลยีและอัลกอริทึมที่สามารถวิเคราะห์ข้อมูลได้อย่างมีประสิทธิภาพ ทำให้มีการพัฒนาการวิเคราะห์ทางด้านภาษาธรรมชาติเป็นไปอย่างก้าวกระโดด

2.3 การจัดการกับข้อมูลที่ไม่สมดุล (SMOTE)

ปัญหา Imbalance Dataset สำหรับการเรียนรู้ของเครื่องนั้นเป็นปัญหาที่พบได้ในกรณีของเทคนิคการจำแนกกลุ่มข้อมูล กล่าวคือปัญหาเกิดจากการมีจำนวนข้อมูลในแต่ละกลุ่ม (Class) ที่ไม่เท่ากัน ซึ่งเป็นหนึ่งในปัจจัยที่มีผลต่อประสิทธิภาพการท างานของแบบจำลอง หากข้อมูลในแต่ละClassมีจำนวนที่ใกล้เคียงกันจะทำให้กระบวนการเรียนรู้ของแบบจำลองมีประสิทธิภาพที่ดีขึ้น แต่หากข้อมูลในแต่ละClass มีจำนวนที่ต่างกันมากอาจทำให้เกิดปัญหาความเหลื่อมล้ำ (Bias) ซึ่งจะส่งผลให้ได้ค่าความแม่นยำสูงในการทำนาย Class ที่มีข้อมูลจำนวนมากและได้ค่าความแม่นยำต่ำใน Class ที่มีจำนวนข้อมูลน้อย ทำให้ผลการทำนายโดยรวมไม่มีประสิทธิภาพเท่าที่ควร [38]

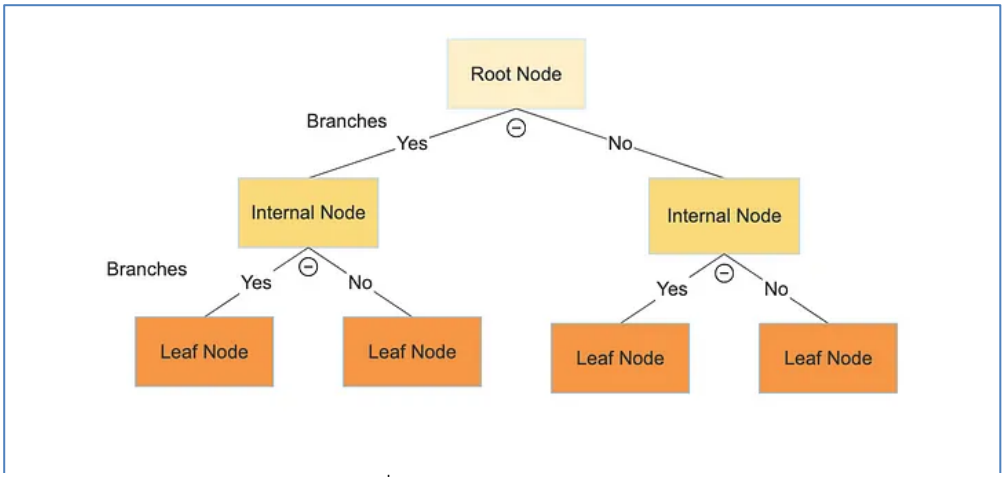
งานวิจัยนี้ใช้วิธีการทำ Sampling Methods ด้วยเทคนิค SMOTE สำหรับการลดปัญหา Imbalance Dataset ซึ่งเป็นเทคนิคที่มีประสิทธิภาพและได้รับความนิยมอย่างมาก มีหลักการทำงานโดยการสุ่มตัวอย่างข้อมูล (Data Point) จาก Class ที่มีข้อมูลจำนวนน้อย (Minority Class) จากชุดข้อมูลเดิม เพื่อเพิ่มจำนวนข้อมูลของ Class นั้นๆ ให้มีจำนวนเท่ากับ Class ที่มีจำนวนข้อมูลมากที่สุดโดยเท่า ๆ กันทุกคลาส

2.4 เทคนิคการจำแนกกลุ่มข้อมูล (Classification)

เป็นเทคนิคหนึ่งของการเรียนรู้ด้วยเครื่องแบบมีผู้สอน เป็นเทคนิคหนึ่งของการเรียนรู้ของเครื่องแบบมีผู้สอน โดยกระบวนการเรียนรู้นั้นจำเป็นต้องมีข้อมูลคำตอบ ซึ่งจะแบ่งข้อมูลคำตอบออกเป็น Class ต่างๆ สำหรับหลักการทำงานของแบบจำลองจะแบ่งข้อมูลออกเป็น 2 ส่วน คือชุดข้อมูลการเรียนรู้และชุดข้อมูลทดสอบซึ่งในงานวิจัยนี้แบ่งข้อมูลดังกล่าวในอัตราส่วน 80 : 20 หรือตามความเหมาะสม จากนั้นนำข้อมูลชุดการเรียนรู้ป้อนเข้าสู่ระบบเพื่อเข้าสู่ขั้นตอนการเรียนรู้โดยใช้อัลกอริทึมที่กำหนด จนกระทั่งได้ผลลัพธ์มาเป็นแบบจำลองการทำนาย จากนั้นนำข้อมูลชุดทดสอบซึ่งเป็นข้อมูลที่เตรียมไว้สำหรับทดสอบประสิทธิภาพการท างานของแบบจำลองมาทำใช้ทำนายผลและพิจารณาผลลัพธ์ที่ได้ โดยในงานวิจัยนี้ได้ใช้เทคนิคในการจำแนกกลุ่มข้อมูล

1) ต้นไม้ตัดสินใจ (Decision tree)

เป็นการใช้โครงสร้างต้นไม้ เพื่อแสดงเส้นทางการตัดสินใจที่นำไปสู่ผลลัพธ์สุดท้าย คือ คำตอบของคำถามที่ตั้งไว้ ในที่นี้ คือ คลาสข้อมูล ต้นไม้ตัดสินใจต้นหนึ่ง ประกอบด้วยองค์ประกอบ คือ โหนด (Node) และกิ่งทางเลือกการตัดสินใจ โดยโหนดสามารถแบ่งได้เป็น 2 ประเภท ได้แก่ โหนดตัดสินใจ (Decision node) ซึ่งใช้แทนคำถามเพื่อการตัดสินใจเลือกกิ่งทางเลือกใดทางเลือกหนึ่งที่เชื่อมโยงจากใต้โหนดประเภทนี้ โหนดการตัดสินใจแรกซึ่งจะถูกพิจารณาก่อนเสมอ คือ โหนดราก (Rood note) จะอยู่บนสุดของต้นไม้ โหนดรากนี้จะเป็นจุดเริ่มต้นของเส้นทางการตัดสินใจข้อแรกของต้นไม้ตัดสินใจ โหนดอีกประเภทหนึ่ง คือ โหนดใบ (Leaf node) ซึ่งเป็นปลายทางของการตัดสินใจ ในกรณีต้นไม้ตัดสินใจสำหรับการวิเคราะห์การจำแนกประเภท โหนดใบนี้ คือ ค่าของตัวแปรเป้าหมายที่เกิดจากการพยากรณ์โดยต้นไม้ติดสินใจ ดังตัวอย่างภาพที่ 2-4

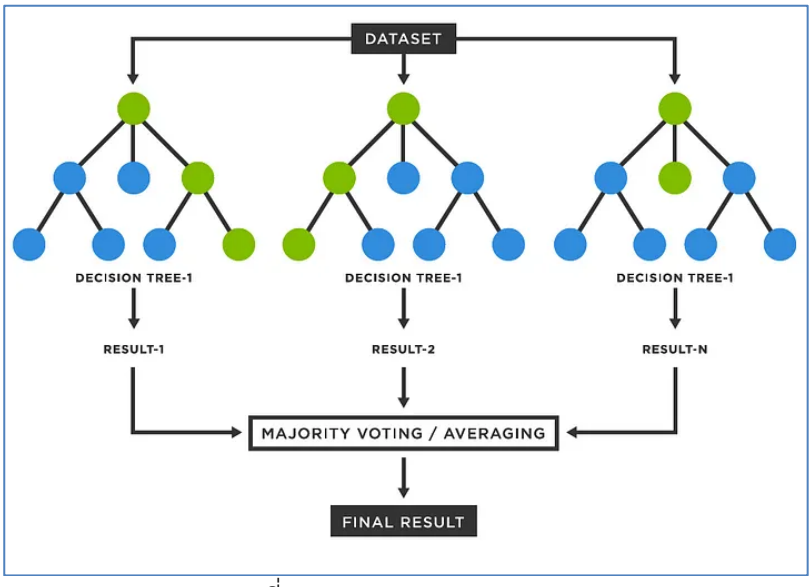


ภาพที่ 2-4 Decision tree Structure

ที่มา: Jainvidip (2021), Understanding Decision Trees: https://medium.com/@jainvidip/understanding-decision-trees-1ba0ef5f6bb4

2) Random forest

เป็นหนึ่งในแบบจำลองที่อยู่ในกลุ่มของ Ensemble Learning ซึ่งมีพื้นฐานมาจากต้นไม้ตัดสินใจ (Decision Tree) ซึ่งจะทำการสร้างแบบจำลอง Decision tree หลาย ๆ ต้น โดยสร้างจากการสุ่มข้อมูลตัวอย่างจากชุดข้อมูลฝึกฝน (Training Data) แบบเลือกแล้วใส่กลับ เพื่อให้มีโอกาสถูกเลือกอีกครั้ง ซึ่งจะสุ่มเลือกข้อมูลให้ได้จำนวน N ตัวอย่าง และสุ่มเลือกแอทริบิวต์เป็นจำนวนที่น้อยกว่าจำนวนแอตทริบิวต์ทั้งหมด โดยแบบจำลองจะมีการทำนายผลออกมาซึ่งจะนำผลออกมาซึ่งจะนำผลการทำนายที่ได้มาโหวตหาผลการทำนายที่ได้รับการโหวตมากที่สุด (Majority Voting) ดังภาพที่ 2-5



ภาพที่ 2-5 Random Forest Structure

ที่มา: Fraidoon Omarza, (2024), Random Forest In-Depth: https://medium.com/@fraidoonomarzai99/random-forest-in-depth-f0556817c40b

3) Naïve Bayes

เป็นตัวแบบสำหรับการจำแนกประเภทที่มีพื้นฐานอยู่บนทฤษฎีของเบย์ (Bayes' Theorem) ซึ่งอธิบายความน่าจะเป็นของการเกิดเหตุการณ์ A เมื่อมีเหตุการณ์ B เกิดขึ้นก่อน P(A|B) ด้วยสมการ (5) คณิตศาสตร์ดังนี้

$$P(A|B) = \frac{P(A)P(B|A)}{P(B)} \quad (5)$$

เมื่อ P(A) และ P(B) คือ ความน่าจะเป็นของการเกิดเหตุการณ์ A และ B ตามลำดับ เราเรียก P(A) ว่าความน่าจะเป็นเบื้องต้น (Prior Probability) ซึ่งมักจะสื่อถึงความเชื่อดั้งเดิมหรือสมมติฐานเก่า เรียก P(B) ว่าหลักฐาน (Evidence) และ P(B|A) คือ ความน่าจะเป็นของการเกิดเหตุการณ์ B เมื่อเหตุการณ์ A เกิดขึ้น เรียกว่า Likelihood ซึ่งมักได้จากข้อมูลจากการสังเกต ส่วน P(A|B) เราเรียกว่า ความน่าจะเป็นภายหลัง (Posterior Probability) ซึ่งเป็นความน่าจะเป็นที่ต้องการทราบค่า ตัวแบบนี้ การทำนายค่าคลาสของข้อมูล x = $(x_1,x_2,,...,x_d)$ เมื่อ x_i คือ ตัวแปรแต่ละตัวแปร โดยการคำนวณค่าความน่าจะเป็นภายหลัง P(l|x) สำหรับแต่ละค่าคลาส $l \in C$ ภายใต้สมมติฐานว่า x_i แต่ละตัวแปรต้น เป็นอิสระต่อกัน ค่า P(l|x) สามารถคำนวณได้จากสมการ (6)

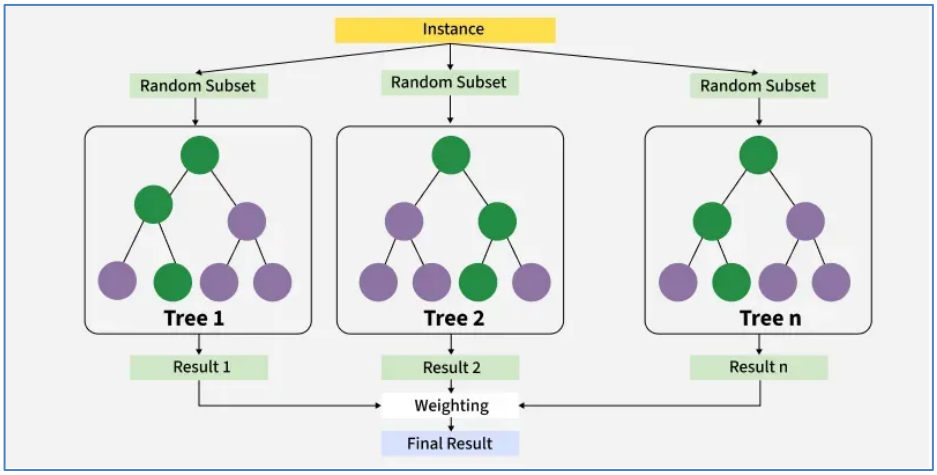
$$P(l|x) = P(l) \prod_{i=1}^{d} P(x_i|l) \quad (6)$$

โดยที่ค่าความน่าจะเป็น P(l) และ P(xi|l) สามารถหาได้โดยวิธีการนับความถี่ของข้อมูลในชุดฝึกสอน ค่าคลาสของข้อมูล x คือ ค่า l ที่ทำให้ค่าความน่าจะเป็นภายหลัง P(l|x) มีค่าสูงที่สุด เราสามารถนิยามตัวแบบเบย์ชนิดนาอีฟในรูปฟังก์ชันคณิตศาสตร์ ได้สมการ (7)

$$f(x) = \frac{argmax P(l|x)}{l \epsilon C} \quad (7)$$

4) Extreme gradient Boosting (XGBoost)

XGBoost เป็นเทคนิคที่พัฒนาจากเทคนิค Gradient Boosting เพื่อเพิ่มความแม่นยำและความยืดหยุ่นให้กับแบบจำลอง โดยใช้หลักการของ Ensemble Learning Method ในการ Boosting เพื่อสร้างตัวเรียนรู้หลาย ๆ ตัว (Multiple Learner) หรือเรียกได้ว่าเป็นการรวม Weak Learners หลายๆตัวเข้าด้วยกัน ซึ่ง Learner ที่สร้างขึ้นใหม่แต่ละรุ่นนั้นจะทำการแก้ไขข้อบกพร่องในการทำงานของ Learner รุ่นก่อนหน้าเพื่อลด Error ดังแสดงในภาพที่ 2-6



ภาพที่ 2-6 การจำลองกลุ่มข้อมูลด้วยเทคนิค XGBoost

ที่มา: https://www.geeksforgeeks.org/machine-learning/xgboost/

5) Long Short-Term Memory (LSTM)

เป็นโมเดลที่ถูกพัฒนาต่อยอดมาจาก RNN (Recurrent Neural Network) โดยเป็นการแก้ปัญหาในส่วนของ Gradient Vanishing ด้วยการออกแบบการทำงานในส่วนของ Cell ใหม่ให้สามารถเก็บสถานะของการคำนวณได้ โดยใน Cell ใหม่ ให้สามารถเก็บสถานะของการคำนวณได้ โดยใน Cell ของ LSTM นั้นมีหน่วยคำนวณย่อย เรียกว่า Grate ซึ่งประกอบด้วย Input Gate, Forget Gate, Memory Cell state gate และ Output Gate

Input Gate เป็นหน่วยย่อยในการกำหนดข้อมูลที่จะนำเข้ามาวิเคราะห์ใน Cell โดยรับข้อมูลเข้ามาเพื่อทำการเขียนค่าลงไปในแต่ละ Cell ดังสมการที่ 8

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}C_{t-1} + b_i) \quad (8)$$

เมื่อ i_t คือ ผลลัพธ์ที่ได้จาก Input Gate

σ คือ ฟังก์ชัน Sigmoid

W_{xf} คือ ค่าน้ำหนักสำหรับคำนวณ Input ใน Forget Gate

x_t คือ Input ที่นำเข้ามาคำนวณ

W_{hf} คือ ค่าน้ำหนักสำหรับคำนวณ Hidden State ใน Forget Gate

h_{t-1} คือ ค่า Hidden State ที่ได้มาจากการคำนวณในหน่วยเวลาก่อนหน้า

W_{hf} คือ ค่าน้ำหนักสำหรับคำนวณ Memory Cell State ใน Forget Gate

C_{t-1} คือ ค่าน้ำ Memory Cell State ที่ได้จากการคำนวณในหน่วยเวลาก่อนหน้า

b_f คือ ค่า Bias ที่ใช้ใน Forget Gate

Memory Cell State Gate เป็นหน่วยย่อยในการกำหนดข้อมูลที่นำเข้ามาวิเคราะห์ ใน Cell และทำการคำนวณค่าสถานะ เพื่อใช้ในการคำนวณในครั้งถัดไป ดังสมการที่ 9

$$c_t = f_t . c_{t-1} + i_t . tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \quad (9)$$

เมื่อ c_t คือค่า Memory Cell state ในช่วงหน่วยเวลา

f_t คือผลลัพธ์ที่ได้จาก Forget Gate

c_{t-1} คือ Memory Cell state จากหน่วยเวลาก่อนหน้า

i_t คือ ผลลัพธ์ที่ได้จาก Input Gate

$tanh$ คือ ฟังก์ชัน Hyperbolic tangent

W_{xc} คือ ค่าน้ำหนักสำหรับคำนวณค่า Input จาก Memory Cell State Gate

x_t คือ ค่า Input ที่นำเข้ามาคำนวณ

W_{hc} คือ ค่าน้ำหนักสำหรับ Hidden State ใน Memory Cell state gate

h_{t-1} คือค่า Hidden State ที่ได้จากการคำนวณในหน่วยเวลาก่อนหน้า

b_c คือค่า Bias ที่ใช้ในการคำนวณใน Forget Gate

Output Gate เป็นหน่วยย่อยสำหรับการคำนวณ Output ของ Cell ซึ่งผลลัพธ์ที่ได้ จาก Cell นี้จะมีอยู่ 2 อย่าง ได้แก่ Output และ Hidden State สำหรับใช้ในการคำนวณครั้งถัดไป โดย มีสมการดังสมการที่ 10 และ 11

$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_{t-1} + b_0 \quad (10)$$

$$h_t = o_t tanh(c_t) \quad (11)$$

เมื่อ o_t คือผลลัพธ์ที่ได้จาก Output Gate

σ คือฟังก์ชัน Sigmoid

W_{xo} คือค่าน้ำหนักสำหรับคำนวณ Input ใน Output Gate

x_t คือค่า Input ที่นำเข้ามาคำนวณ

W_{ho} คือค่าน้ำหนักสำหรับคำนวณ Hidden State ใน Output Gate

h_{t-1} คือค่า Hidden State ที่ได้มาจากการคำนวณในหน่วยเวลาก่อนหน้า

W_{co} คือค่า น้ำหนักสำหรับคำนวณ Memory Cell state gate ใน Output Gate

c_{t-1} คือค่า Memory Cell state ที่ได้จากการคำนวณในหน่วยเวลาก่อนหน้า

b_0 คือค่า Bias ใช้ในการคำนวณใน Output Gate

h_t คือค่า Hidden State จากการคำนวณ

2.5 การวัดประสิทธิภาพของโมเดล

ในส่วนของการทำนายโมเดลการจำแนกประเภท (Classification) ในงานวิจัยนี้จะใช้ค่า Accuracy, Precision, Recall และ F1-score ในการวัดประสิทธิภาพของโมเดล

1) Accuracy คือ การวัดค่าความถูกต้องโดยรวมของระบบระหว่างค่าจริงและค่าการทำนาย ถ้าหากค่า Accuracy มีค่ามาก นั่นหมายถึง ค่าการทำนายนั้นสามารถทำนายได้ถูกต้องใกล้เคียงกัลค่าจริง ดังสมการ (12)

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (12)$$

เมื่อ Accuracy คือ ค่าความถูกต้อง

TP คือ ค่าการทำนายที่ทำนายว่าจริงซึ่งตรงกับค่าจริง (True Positive)

TN คือ ค่าการทำนายที่ทำนายว่าไม่จริงซึ่งตรงกับค่าจริง (True Negative)

FP คือ ค่าการทำนายที่ทำนายว่าไม่จริงซึ่งไม่ตรงกับค่าจริง (False Positive)

FN คือ ค่าการทำนายที่ทำนายว่าจริงซึ่งไม่ตรงกับค่าจริง (False Negative)

2) Precision คือการวัดค่าความแม่นยำโดยวัดจากความซ้ำเดิมของค่าการทำนายที่ทำนายได้ถูกต้องตรงกับค่าจริง หากค่า Precision มีค่ามาก นั่นหมายถึงค่าการทำนายนั้นสามารถ ทำนายได้แม่นยำใกล้เคียงกับค่าจริง ดังสมการที่ 13

$$\text{Precision} = \frac{TP}{TP+FP} \quad (13)$$

เมื่อ Precision คือ ค่าความแม่นยำ

TP คือ ค่าการทำนายที่ทำนายว่าจริงซึ่งตรงกับค่าจริง (True Positive)

FP คือ ค่าการทำนายที่ทำนายว่าไม่จริงซึ่งไม่ตรงกับค่าจริง (False Positive)

3) Recall คือการวัดค่าความครบถ้วน ซึ่งหมายถึงอัตราส่วนการวัดค่าการทำนายที่ทำนายได้ถูกต้องตรงกับค่าจริงจากจำนวนของค่าจริงทั้งหมด หากค่า Recall มีค่ามาก นั่นหมายถึงค่าการทำนายนั้นสามารถทำนายได้อย่างครบถ้วนใกล้เคียงกับค่าจริง ดังสมการที่. (14)

$$\text{Recall} = \frac{TP}{TP+FN} \quad (14)$$

เมื่อ Recall คือ ค่าความครบถ้วน

TP คือ ค่าการทำนายที่ทำนายว่าจริงซึ่งตรงกับค่าจริง (True Positive)

FN คือ ค่าการทำนายที่ทำนายว่าจริงซึ่งไม่ตรงกับค่าจริง (False Negative)

4) F1-score คือ ค่าเฉลี่ย Harmonic ระหว่างค่า precision และ recall เพื่อใช้ในการวัดความสามารถของโมเดล โดยเฉพาะโมเดลที่มีข้อมูลไม่สมดุล (Imbalance Data) คำนวณดังสมการที่ 15

$$\text{F}_1 - \text{score} = \frac{2\times(\text{PrecisionxRecall})}{(\text{Precision}+\text{Recall})} \quad (15)$$

เมื่อ F1-score คือ ค่าประสิทธิภาพเฉลี่ย Harmonic

Precision คือ ค่าความแม่นยำ

Recall คือ ค่าความครบถ้วน

2.6 งานวิจัยที่เกี่ยวข้อง

1) งานวิจัยต่างประเทศ

งานวิจัยของ Xuan และคณะ [39] กล่าวถึง Random Forest เป็นหนึ่งในวิธีในการนำมาใช้ในการตรวจจับการฉ้อโกงของการใช้งานบัตรเครดิต โดย Random Forest จะใช้ในการจำแนกประเภท ซึ่งเป็นอัลกอริทึมที่นิยมในการทำต้นไม้ตัดสินใจ เนื่องจากมีความยืดหยุ่นในการจัดการคุณลักษณะข้อมูลประเภทต่างๆ อย่างไรก็ตามแบบจำลองต้นไม้ตัดสินใจต้นเดียวอาจมีประสิทธิภาพที่ไม่เพียงพอและทำให้เกิดการ Overfit ซึ่งจะใช้เทคนิค Ensemble ในการแก้ปัญหาเหล่านี้ โดยการรวมกลุ่มกันของต้นไม้ตัดสินใจหลาย ๆ ต้น จะช่วยเพิ่มความแม่นยำมากกว่าการท าต้นไม้ตัดสินใจต้นเดียว ซึ่งข้อดีของ Random Forest คือมีความแข็งแกร่งต่อสัญญาณรบกวน (noise) และมีความแข็งแกร่งต่อค่าที่ผิดปกติ (outlier)

งานวิจัยของ Wang และคณะ [40] กล่าวถึง XGBoost เป็นอัลกอริทึมที่ถูกพัฒนาขึ้นมาจาก Gradient Tree Boosting ซึ่งสามารถจัดการงานที่มีข้อมูลขนาดใหญ่ได้อย่างมีประสิทธิภาพ XGBoost เป็นอัลกอริทึมที่มีประสิทธิภาพ และใช้เวลา ในการพัฒนาแบบจำลองที่ไม่นาน ซึ่งมีการประยุกต์ใช้งานในด้านการวิจัยที่หลากหลายตั้งแต่การวินิจฉัยโรคมะเร็งไปจนถึงการประเมินความเสี่ยงด้านการใช้งานบัตรเครดิต ซึ่งในปัจจุบัน XGBoost ได้กลายเป็นวิธีการทางเลือกแรกสำหรับใช้ในการพัฒนาแบบจ าลองของข้อมูลขนาดใหญ่ และยังเป็นวิธีการในการพัฒนาแบบจำลองที่ได้รับความนิยมมากที่สุด ถึงแม้ว่า XGBoost ประสบความสำเร็จอย่างมาก แต่ประสิทธิภาพของมันมักจะลดลง เมื่อชุดข้อมูลที่ถูกนำมาใช้ในการวิเคราะห์มีปัญหาความไม่สมดุลกันของชุดข้อมูล (Imbalance Data) แต่มีหลายงานวิจัยที่บอกว่า XGBoost สามารถใช้ในการจัดการกับปัญหาความไม่สมดุลกันของชุดข้อมูลได้ดี ซึ่งสามารถทำงานได้อย่างมีประสิทธิภาพเหนือกว่าวิธีการอื่นๆ ในการจัดการกับปัญหาความไม่สมดุลกันของชุดข้อมูล ซึ่งในงานวิจัยนี้ได้มีการแนะนำ imbalance-XGBoost ซึ่งเป็นแพ็คเกจ Python ที่ใช้ XGBoost ในการแก้ไขปัญหาความไม่สมดุลกันของชุดข้อมูล

งานวิจัยของ Nguyen และคณะ [41] เสนอแนวทางในการสร้างระบบสำหรับการคาดการณ์อาชีพโดยนำแบบจำลองต้นไม้การตัดสินใจแบบ Extreme Gradient Boosting (XGBoost)มาใช้กับผลการเรียนของบัณฑิตจากคณะวิทยาการคอมพิวเตอร์และวิศวกรรมศาสตร์ของมหาวิทยาลัยนานาชาติโฮจิมินห์ในช่วง 5 ปีที่ผ่านมา (ปี 2014 ถึง 2018) จำนวน 177 คน โดยเป็นข้อมูล รหัสประจำตัวของผู้สำเร็จการศึกษา รหัสประจำตัว ชื่อ และจำนวนหน่วยกิต ภาคการศึกษาและปีที่เรียนมีการทดสอบ 2 ชุดย่อยชุดหนึ่งสำหรับการฝึกอบรม (80 เปอร์เซ็นต์) และอีกชุดหนึ่งสำหรับการทดสอบ (20 เปอร์เซ็นต์) โดยมีค่า Accuracy เท่ากับ 87.5%

Repaso และ Caparino [42] ได้เก็บรวบรวมข้อมูลเชิงสำรวจและเทคนิคการจำแนกประเภทเพื่อทำนายความเชี่ยวชาญด้านอาชีพของบัณฑิตในสาขาเทคโนโลยีสารสนเทศ และนำมาวิเคราะห์ด้วยเทคนิค Machine learning โดยโมเดล Naïve Bayes และ Random Forest แสดงให้เห็นถึงความแม่นยำที่สูงขึ้น อย่างไรก็ตาม ผลการศึกษาอาจไม่สามารถสรุปผลได้เกินขอบเขตของข้อมูลบัณฑิตสาขาเทคโนโลยีสารสนเทศ

ของ Bulacan State University Sarmiento จำเป็นต้องมีการวิจัยเพิ่มเติมเพื่อสำรวจการแทรกแซงเพื่อปรับปรุงผลลัพธ์ของนักศึกษาในหลักสูตรอื่นๆ

Kamal และคณะ [43] ได้นำเสนอ A Smart Career Guidance System ที่ช่วยให้นักศึกษาที่กำลังจะเริ่มต้นการศึกษาระดับสูงสามารถวิเคราะห์ทักษะ ความสามารถ และความสนใจของตนเอง และแนะนำ 5 สาขาที่เหมาะสมที่สุดด้วยเทคนิค Machine learning จากข้อมูลผู้สำเร็จการศึกษา 392 คนที่ตอบแบบสำรวจออนไลน์นี้ และใช้การสุ่มตัวอย่างแบบ SMOTE พบว่า XGBoost และ Random Forest แนะนำตัวเลือกอาชีพที่เหมาะสมที่สุดได้

Ali และคณะ [44] นำเสนอแนวทางการใช้ Natural Language Process เพื่อทำนายหมวดหมู่งานจากประวัติย่อ (Resume) ของผู้สมัครงาน การศึกษานี้ใช้ชุดข้อมูล Resume ทั้งหมด 962 รายการในแต่ละ 25 หมวดหมู่ของงาน และประเมินโมเดลการจำแนกประเภทของ Machine learning โมเดล ได้แก่ SVM, Naïve Bayes และ Logistic Regression ผลลัพธ์แสดงให้เห็นว่าตัวจำแนก SVM โดยเฉพาะตัวจำแนก Linear Support Vector มีความแม่นยำมากกว่า 96%

Germán และคณะ [45] ประยุกต์ใช้ Machine learning ได้แก่ Support Vector Machines, Naive-Bayes, Logistic Regression, Random Forest, and deep learning Long-Short Term Memory (LSTM) สำหรับจำแนกประเภทงานทั้งหมด 23 กลุ่มงาน ได้แก่ ได้แก่ การขาย การบริหาร ศูนย์บริการทางโทรศัพท์ เทคโนโลยี การค้า ทรัพยากรบุคคล โลจิสติกส์ การตลาด สุขภาพ อาหาร การเงิน เลขานุการ การผลิต วิศวกรรม การศึกษา การออกแบบ กฎหมาย การก่อสร้าง การประกันภัย การสื่อสาร การจัดการ การค้าต่างประเทศ และการทำเหมืองแร่ โดยใช้ SMOTE, Geometric-SMOTE, and ADASYN เพื่อจัดการกับข้อมูลที่ไม่สมดุลซึ่งเครือข่ายประสาทเทียมแบบคอนโวลูชั่นให้ผลลัพธ์ที่ดีที่สุดเมื่อใช้อัลกอริทึม Geometric-SMOTE

2) งานวิจัยในประเทศไทย

จามิกรและอติชาต [46] ได้ศึกษาเพื่อระบุทักษะทางเทคนิค (hard skills) และทักษะด้านบุคคล (soft skills) ที่เป็นที่ต้องการในอุตสาหกรรมซอฟต์แวร์ของประเทศไทย โดยใช้เทคนิค Text Mining บนประกาศรับสมัครงานออนไลน์ โดยรวบรวมข้อมูลจากแหล่งประกาศงานสำคัญ เช่น JobThai, JobDB, LinkedIn แล้วประมวลผลด้วย Text Mining เพื่อดึงและวิเคราะห์ทักษะที่ปรากฏในประกาศงานเหล่านั้น ผลการวิจัย พบว่าทักษะทางเทคนิคที่เป็นที่ต้องการสูงประกอบด้วยภาษาการเขียนโปรแกรม (เช่น Java, SQL) และเทคโนโลยีอื่นๆ และในส่วนของทักษะบุคคล ทักษะที่โดดเด่น ได้แก่ การสื่อสาร การทำงานเป็นทีม และการแก้ปัญหา ซึ่งสอดคล้องกับงานวิจัยอื่น ๆ ผลงานนี้มีแนวทางไปสู่การออกแบบหลักสูตรที่ตอบโจทย์ตลาดงานมากขึ้น ช่วยให้นักศึกษาพร้อมมากขึ้นสำหรับการทำงานจริง รวมถึงการใช้โมเดลทำนายเชิงทักษะเพื่อแนะนำทักษะที่ควรพัฒนา

ธนาธารและมนต์ชัย [47] ทำการสำรวจ งานวิจัยที่เกี่ยวข้องกับการจัดประเภทข้อความอัตโนมัติ (Automatic Text Classification) โดยใช้ข้อมูลจากโซเชียลมีเดียในรูปแบบภาษาไทย ได้แก่ เช่น Facebook, Twitter และ Pantip เพื่อเปรียบเทียบเทคนิค Naïve Bayes, SVM, Decision Treesม, Random Forest, kNN และ Deep Learning เช่น CNN, LSTM, RNN และการใช้ Feature Representation เช่น Bag-of-

Words, TF-IDF และ Word Embedding ผลการสำรวจพบว่า ภาษาไทยมีความซับซ้อนในการวิเคราะห์กว่าภาษาอังกฤษ เช่น การไม่มีการเว้นวรรคระหว่างคำ และความหลากหลายของรูปคำ งาน Text Classification ภาษาไทยยังมีข้อจำกัด ด้านฐานข้อมูลมาตรฐาน การทำ preprocess และ tokenization โมเดล Deep Learning เช่น LSTM หรือ CNN เริ่มมีบทบาทมากขึ้นในงานวิเคราะห์ข้อความภาษาไทย ขาดการเปรียบเทียบข้ามโมเดลแบบ systematic และยังไม่มี benchmark ที่ชัดเจนในหลายโดเมนของข้อความไทย

พยุง [48] ได้เสนอกรอบการทำงานสำหรับการตรวจจับข่าวปลอมในประเทศไทย ประกอบด้วยโมดูลหลักสามโมดูล ได้แก่ การดึงข้อมูล การประมวลผลภาษาธรรมชาติ และการเรียนรู้ของเครื่อง การวิจัยนี้มีสองขั้นตอน ได้แก่ ขั้นตอนการรวบรวมข้อมูลและขั้นตอนการสร้างแบบจำลองการเรียนรู้ของเครื่อง ในขั้นตอนการรวบรวมข้อมูล เราได้รวบรวมข้อมูลจากเว็บไซต์ข่าวออนไลน์ของไทยโดยใช้การดึงข้อมูลจากเว็บไซต์และวิเคราะห์ข้อมูลโดยใช้เทคนิคการประมวลผลภาษาธรรมชาติเพื่อดึงข้อมูลที่ดีจากข้อมูลบนเว็บ เพื่อการเปรียบเทียบ เราได้เลือกแบบจำลองการเรียนรู้ของเครื่องที่เป็นที่รู้จัก ได้แก่ Naïve Bayesian, Logistic Regression, K-Nearest Neighbor, Multilayer Perceptron, Support Vector Machine, Decision Tree, Random Forest, Rule-Based Classifier และ Long Short-Term Memory การศึกษาเปรียบเทียบชุดทดสอบแสดงให้เห็นว่า LSTM เป็นแบบจำลองที่ดีที่สุด

บทที่ 3
วิธีการดำเนินการวิจัย

วิธีการดำเนินการวิจัยครั้งนี้ ผู้วิจัยแบ่งการดำเนินการออกในการเก็บรวบรวมข้อมูลทักษะของวิศวกรรมซอฟต์แวร์เป็น 2 กลุ่ม 1) ศึกษาสมรรถนะตามเอกสารและงานวิจัยที่เกี่ยวข้องกับวิศวกรรมซอฟต์แวร์ทั้งในและต่างประเทศ 2) ศึกษาสมรรถนะของวิศวกรรมซอฟต์แวร์จากประกาศรับสมัครงานขององค์กรซอฟต์แวร์ในประเทศไทย และ 3) สร้างตัวแบบสำหรับประเมินตำแหน่งงานที่เหมาะกับสมรรถนะในตำแหน่งงานที่กำหนด โดยมีรายละเอียดแต่ละขั้นตอนดังนี้

3.1 ศึกษาสมรรถนะตามเอกสารและงานวิจัยที่เกี่ยวข้องกับวิศวกรรมซอฟต์แวร์ทั้งในและต่างประเทศ

1) รวบรวมข้อมูลแหล่งข้อมูลที่ใช้สำหรับการศึกษาในขั้นตอนนี้ประกอบแหล่งข้อมูลในประเทศไทยและต่างประเทศที่เกี่ยวข้องกับตำแหน่งงานของวิศวกรรมซอฟต์แวร์ ได้แก่

1.1) แหล่งข้อมูลที่เกี่ยวข้องกับข้อกำหนดสมรรถนะวิศวกรรมซอฟต์แวร์ ได้แก่

- Software Engineering Body of Knowledge version 3.0 (SWEBOK v.3)
- The global skills and competency framework for the digital world (SFIA Ver.8)
- Software Engineering Competency Model (SWECOM)
- HRSG's Competency Dictionary 2020
- Information Technology Competency Model 2017
- Terminology of European education and training policy

1.2) รวบรวมเอกสารต่างๆ ด้วยการดาวน์โหลดข้อมูลจากแหล่งข้อมูลที่กำหนดไว้ และฐานข้อมูลออนไลน์ต่างๆ Google Scholar และ IEEE ซึ่งเป็นบทความวิจัยที่เกี่ยวข้องระหว่างปี 2020-2024

1.3) วิเคราะห์ข้อมูล โดยใช้วิธีการวิเคราะห์เนื้อหา จากนั้นจัดกลุ่มข้อมูลตำแหน่งงานของวิศวกรรมซอฟต์แวร์ 7 ตำแหน่งงาน ประกอบด้วย

1. วิศวกรความต้องการ (Requirements Engineer)
2. นักออกแบบซอฟต์แวร์ (Software Designer)
3. นักพัฒนาซอฟต์แวร์ (Developer)
4. นักทดสอบคุณภาพซอฟต์แวร์ (Software Tester)
5. วิศวกรซ่อมบำรุง (Maintenance Engineer)
6. ผู้จัดการโครงการซอฟต์แวร์ (Project Manager)
7. นักวิเคราะห์ระบบ (System Analyst)

2) วิเคราะห์สมรรถนะที่พึงมีตามการรวบรวมเอกสารและงานวิจัยที่เกี่ยวข้อง ในขั้นตอนนี้ทำให้ได้กลุ่มคำของสมรรถนะที่สอดคล้องกับตำแหน่งงานในข้อที่ 1.3) ประกอบด้วย 2 กลุ่มทักษะได้แก่

2.1) ทักษะด้านเทคนิค (Hard skill) แบ่งออกเป็น 6 กลุ่มทักษะย่อย ได้แก่ 1) ภาษาการเขียนโปรแกรม (Programming languages) 2) แพตฟอร์ม (Platform) การจัดเก็บและฐานข้อมูล (Database and Data Usage) 3) เฟรมเวิร์ก (Framework) 4) เครื่องมือสำหรับพัฒนา (Development tools และ 5) เครื่องมือทดสอบ (Testing tools)

2.2) ทักษะด้านบุคคล (Soft skill) จำนวน 36 คำ เช่น Activeness, Analytical, Collaborate, Commitment, Communication, Conceptual skills, Creative, Critical thinking, Decision making, Customer orientation เป็นต้น [4-14]

3.2 ศึกษาสมรรถนะของวิศวกรรมซอฟต์แวร์จากประกาศรับสมัครงาน

ในขั้นตอนนี้ผู้วิจัยได้ประยุกต์ใช้เทคนิค NLP ร่วมกับการทำเหมืองข้อความ (Text mining) โดยมีรายละเอียดของวิธีการดำเนินการ ดังนี้

1) แหล่งข้อมูล

ผู้วิจัยเลือกใช้ข้อมูลจากกลุ่มผู้ใช้และองค์กรบนสื่อสังคมออนไลน์แบบไม่ระบุตัวตน ซึ่งระบุถึงความต้องการด้านสมรรถนะของวิศวกรซอฟต์แวร์ ผ่านประกาศรับสมัครงานจากเว็บไซต์ที่เปิดเผยต่อสาธารณะ ได้แก่ JobThai (http://jobthai.com) และ LinkedIn (https://www.linkedin.com/) โดยเว็บไซต์ดังกล่าวมีสัญญาอนุญาตที่สามารถเข้าถึงข้อมูลหน้าเว็บไซต์ได้ ซึ่งการดำเนินการเก็บข้อมูลดำเนินการภายใต้เงื่อนไขที่ไม่ขัดต่อข้อกำหนดการใช้งานของเว็บไซต์ดังกล่าว

2) เก็บรวบรวมข้อมูลและการเตรียมข้อมูล

การเก็บรวบรวมข้อมูลดำเนินการผ่านกระบวนการเหมืองข้อความ (Text Mining) โดยแบ่งออกเป็น 3 ขั้นตอนหลัก ดังนี้

2.1) รวบรวมข้อมูลจากเว็บไซต์ ดำเนินการรวบรวมข้อมูลผ่านวิธีการดึงข้อมูลหน้าเว็บไซต์ (Web Scraping) โดยใช้วิธีที่เหมาะสมกับลักษณะโครงสร้างของเว็บไซต์ ดังนี้

1. เว็บไซต์ที่ใช้เทคโนโลยี Server-Side Rendering จะใช้ภาษา JavaScript โดยผ่าน Visual Studio Code โดยติดตั้ง Node.js และแพ็กเกจ Cheerio ซึ่งเป็นไลบรารีสำหรับจัดการกับ HTML Document โดยมีลักษณะการทำงานคล้าย jQuery สามารถใช้ในการเลือก DOM Element ได้อย่างสะดวก ในกรณีของเว็บไซต์ประเภทนี้ เมื่อมีการร้องขอข้อมูลผ่าน HTTP Request ข้อมูล HTML ทั้งหมดจะถูกประมวลผลจากฝั่งเซิร์ฟเวอร์ก่อนแล้วส่งมายังฝั่งผู้ใช้งาน ทำให้สามารถดึงข้อมูลที่ต้องการได้ทันทีจากโครงสร้าง HTML

จากนั้นใช้ Cheerio ทำการแปลง HTML ที่ได้รับเป็น DOM Model เพื่อใช้ CSS Selector หรือ XPath สำหรับระบุตำแหน่งของข้อมูลที่ต้องการ และทำการแปลงข้อมูลให้อยู่ในรูปแบบ JSON เพื่อใช้ในการประมวลผลหรือวิเคราะห์ในขั้นตอนถัดไป

2. เว็บไซต์ที่ใช้เทคโนโลยี Client-Side Rendering จะใช้ภาษา JavaScript โดยผ่าน Visual Studio Code เช่นกัน โดยติดตั้ง Node.js และแพ็กเกจ Puppeteer ซึ่งเป็นไลบรารีที่ใช้ควบคุม Web Browser (เช่น Chromium หรือ Chrome) แบบ Headless โดย Puppeteer จะจำลองการเปิดเว็บเพจเหมือนผู้ใช้งานจริง เพื่อให้ JavaScript ภายในหน้าเว็บทำงานจนโหลดข้อมูลและแสดงผล DOM อย่างสมบูรณ์

ในกรณีของเว็บไซต์ประเภทนี้ การดึงข้อมูลด้วยวิธีปกติผ่าน HTTP Request จะไม่สามารถเข้าถึงข้อมูลที่ต้องการได้ทันที เพราะเนื้อหาในหน้าเว็บจะถูกเรนเดอร์โดย JavaScript หลังจากโหลดหน้าเว็บ ทำให้ต้องใช้ Puppeteer เพื่อรอให้ DOM โหลดเสร็จ จากนั้นจึงดึงข้อมูลผ่าน CSS Selector หรือ XPath และจัดเก็บข้อมูลในรูปแบบ JSON เช่นเดียวกัน

2.2) การทำความสะอาดข้อมูล (Data cleaning) การดำเนินงานในขั้นตอนนี้เป็นการเตรียมข้อมูล โดยมีขั้นตอนคือ

1. คัดเลือกเฉพาะประกาศรับสมัครงานที่เป็นภาษาอังกฤษ โดยลบประกาศรับสมัครงานที่บรรยายคุณสมบัติเป็นภาษาไทยออก

2. ลบข้อมูลประกาศงานที่ซ้ำซ้อน (Duplication) ซึ่งเป็นตำแหน่ง บริษัท และรายละเอียดคุณสมบัติเดียวกันออก

3. ลบประกาศตำแหน่งงานที่ไม่เกี่ยวข้องกับตำแหน่งงานที่ผู้วิจัยกำหนดไว้ออก

4. ลบข้อความที่ไม่เกี่ยวข้องกับรายละเอียดคุณสมบัติของตำแหน่งงานนั้น เช่น ข้อมูลที่บรรยายเกี่ยวกับบริษัท หรือข้อมูลการติดต่อ

5. ทำการประมวลผลด้วยโทเค็น (Tokenization) โดยแบ่งส่วนข้อความหรือคำศัพท์ด้วยการแยกข้อมูลออกเป็นคำเล็ก ๆ

ตัวอย่างเช่น เรามีประโยค “Ross 128 is earth like planet.Can we survive in that planet?” หลังจาก tokenization ประโยคนี้จะกลายเป็น

[‘Ross’, ‘128’, ‘is’, ‘earth’, ‘like’, ‘planet’, ‘.’, ‘Can’, ‘we’, ‘survive’, ‘in’, ‘that’, ‘planet’, ‘?’]

tokenization ใน python สามารถทำได้โดยใช้ฟังก์ชัน word_tokenize () ซึ่งเป็น Library NLTK ในภาษา Python

2.3) ปรับคำให้เป็นตัวอักษรพิมพ์เล็กและลบข้อมูลบางส่วน (Noisy data) ที่ไม่เกี่ยวข้องกับทักษะของวิศวกรรมซอฟต์แวร์ โดยเริ่มจากตัวเลข (Number) เครื่องหมายวรรคตอน (Punctuation) ช่องว่าง

ระหว่างคำ (Strip write space) คำหยุด (Stop word) และคำที่ไม่เกี่ยวข้องกับทักษะของวิศวกรรมซอฟต์แวร์ ขั้นตอนนี้ดำเนินการลบโดย Library BeautifulSoup ในภาษา Python ดังตัวอย่าง 3-1

ตารางที่ 3-1 ตัวอย่างคำที่ไม่เกี่ยวข้อง (Remove word)

year, age, degree, education, gender, branch, military, start, consider, special, other, driver's license, body, strong, male, female, salary, month, day, student, week, insurance, bonus, apply, training, number, contact, ot

2.4) จัดข้อมูลที่มีความหมายเหมือนกันให้เป็นข้อมูลเดียวกัน ทำการตรวจสอบคำ ลดความซ้ำซ้อนของคำโดยคำที่มีความหมายเดียวกัน/สอดคล้องกัน/คล้ายคลึงกัน เพื่อกำหนดคำศัพท์ใหม่ในการเป็นตัวแทนหรือข้อความนั้น เช่น Microsoft SQL Server", "MS SQL", "MSSQL Server"จะทำการแทนที่ด้วย MSSQL เป็นต้น ตัวอย่างดังตาราง 3-2

ตารางที่ 3-2 ตัวอย่างการจัดการข้อความที่มีความสอดคล้องกัน

#	Role	Keywords
1	Requirements Engineer (RE)	Requirement Analyst, Business Analyst, Business Systems Analyst, Requirements Engineer, Product Analyst, Functional Analyst
2	Software Designer (SD)	UX/UI Designer, Graphics Engineer, Solution Architect, Technical Architect, Software Architect, Application Architect, UX Architect, Software Designer
3	Developer (Dev)	Programmer, Software Engineer, Full Stack Developer, Frontend Developer, Backend Developer, Web Developer, Mobile Developer, Embedded Developer, DevOps Engineer, Developer
4	Software Tester (ST)	Software Quality Engineer, Software Test Engineer, Quality Assurance Engineer, QA Engineer, Test Automation Engineer, Quality Analyst, Test Lead, Test Manager, Quality Assurance Manager, Release Engineer, Software Tester
5	Maintenance Engineer (ME)	Configuration Management Engineer, Systems Configuration Engineer, Infrastructure Configuration Engineer, Network Configuration Engineer, IT Configuration, Maintenance Engineer: Software Configuration Engineer, Release and Configuration Engineer, Technical Support Engineer, Maintenance Engineer

#	Role	Keywords
6	Project Manager (PM)	Technical Project Manager, Software Project Manager, Scrum Master, Agile Coach, Team Lead, Technical Lead, Project Manager
7.	System Analyst (SA)	System Designer, Software Analyst, Solution Architect, Enterprise Architect, Process Analyst, Information Systems Designer, Technical Analyst

3) พัฒนาพจนานุกรม (Dictionary) เพื่อรองรับการวิเคราะห์ความสอดคล้องกับตำแหน่งงานที่กำหนด ทำให้ได้ชุดทักษะ ทักษะด้านเทคนิค (Hard skill) ซึ่งแบ่งเป็นกลุ่มคือ ทักษะด้านเทคนิค (Technical skill) ดังตารางที่ 3-3 และรายการทักษะเชิงบุคคล (Soft skill) ดังตารางที่ 3-4

ตารางที่ 3-3 ทักษะด้านเทคนิค (Technical skill)

Skill	Sub-skill
Programming languages	C++, C#, Java, Javascript, Nodejs, Python, Ruby, Swift, PHP, Kotlin, Objective-C, TypeScript, Go, Rust, HTML, CSS, SQL, Shell scripting, R, Perl, Scala, Groovy, Dart, Haskell, Lua, Assembly, Shel, VBA etc.
Platform	Operating System Platforms: Windows, macOS, Linux, Android etc. Web development platform: Node.js etc. Desktop Application Platforms: Windows Forms, WPF, JavaFX etc. Game Development Platforms: Unity, Unreal Engine etc. Cloud Platforms: Amazon Web Services (AWS), Microsoft Azure, Google Cloud Platform (GCP) etc.
Database & Data usage	MySQL, PostgreSQL, Microsoft SQL Server, Oracle Database MongoDB, Cassandra, CouchDB, Redis, Memcached, Neo4j, Amazon, Neptune, Apache Cassandra, HBase, InfluxDB, Prometheus, Google Spanner, CockroachDB, ORM technologies etc.

Skill	Sub-skill
Framework	Spring, Spring Boot, Django, Laravel, Express.js, JavaSeverFace (JSF), Wicket, GWT, Spring MVC, Django and Flask, Ruby on Rails etc.
Development tools	Visual Studio, Eclipse, PyCharm, Xcode, Visual Studio code, Git, Github, Gitlab, Bitbucket, Jira, Docker, Kubernetes, Jenkins etc.
Testing tools	JUnit, Selenium, Pytest, Robotium, Katalon, Maven, Selendroid, Linux Desktop, Testing Project, OpenTest, Apache JMeter, Watir, QTest etc.

ตารางที่ 3-4 กลุ่มคำของทักษะด้านบุคคล (Soft skill)

#	Soft skill		Soft skill	#	Soft skill
1.	Activeness	13.	Essence	25.	Research
2.	Analytical	14.	Estimate timeline	26.	Responsibility
3.	Collaborate	15.	Flexibility	27.	Result orientation
4.	Commitment	16.	innovation	28.	Self Management
5.	Communication	17.	Knowledge transfer	29.	Self-confidence
6.	Conceptual skills	18.	Leadership	30.	Self-esteem
7.	Creative	19.	Lift long learning	31.	Self-organization
8.	Critical thinking	20.	Motivation	32.	Situation awareness
9.	Curiosity	21.	Positive attitude	33.	social competence
10.	Customer orientation	22.	Pressure	34.	Teamwork
11.	Decision making	23.	Problem-solving	35.	Visionary
12.	Ethical/work ethic	24.	Quality control	36.	Willingness

โดยการจัดกลุ่มข้อความตามตารางที่ 3-2, ตารางที่ 3-3 และ ตารางที่ 3-4 จะผ่านการคัดกรองจากทีมผู้วิจัยประกอบกับการพิจารณาความสอดคล้องกับงานวิจัยที่ผ่านมา และผ่านการตรวจสอบจากผู้เชี่ยวชาญซึ่งเป็นทีมที่ปรึกษาของผู้วิจัย จึงทำให้การจัดกลุ่มดังกล่าวมีความเชื่อถือได้

4) การวิเคราะห์ข้อมูลคำที่ได้ที่เกิดจากการเก็บรวบรวมจากงานวิจัยที่เกี่ยวข้องกับแหล่งข้อมูลจากเว็บไซต์ประกาศรับสมัครงานเพื่อวิเคราะห์ข้อความ (Text analysis) ตามตำแหน่งงานที่กำหนดวิเคราะห์ด้วยความถี่และร้อยละ

5) สร้างตัวแบบการทำนายตำแหน่งงานตามกลุ่มทักษะตามอาชีพ ด้วยเทคนิคการจำแนกประเภท (Classification techniques) ได้แก่ โดยใช้ตัวแบบในกลุ่ม Machine learning วิธี Random Forest, Naïve Bayes Support, XGBoost และการวิเคราะห์ LSTM ซึ่งอยู่ในกลุ่มการเรียนรู้เชิงลึก (Deep Learning)

ในการสร้างโมเดลทำนายตำแหน่งงานจากทักษะที่ปรากฏในประกาศรับสมัครงาน ผู้วิจัยใช้ชุดข้อมูลจำนวนทั้งสิ้น 337,454 รายการ แบ่งออกเป็น 7 ประเภทของตำแหน่งงาน (จำนวน 7 คลาส) ได้แก่ Requirements Engineer, Software Designer, Developer, Software Tester, Maintenance Engineer, Project Manager และ System Analyst ซึ่งเป็นกลุ่มตำแหน่งงานที่ใช้ในการฝึกและทดสอบโมเดล และเนื่องจากข้อมูลมีความไม่สมดุลระหว่างคลาส ผู้วิจัยจึงใช้เทคนิค SMOTE (Synthetic Minority Over-sampling Technique) เพื่อสร้างตัวอย่างใหม่ให้กับคลาสที่มีจำนวนน้อย โดยใชคลาสของ roject Manager เป็นฐาน โดยพิจารณาที่อัตราส่วน 1:40 (เป้าหมาย = 74,720) เพื่อให้แต่ละตำแหน่งมีความสมดุลมากขึ้น ซึ่งช่วยลด bias ของโมเดลในการทำนาย โดยมีข้อมูลที่นำไปใช้ในการวิเคราะห์ ดังตารางที่ 3-5

ตารางที่ 3-5 จำนวนประกาศรับสมัครงานก่อนและหลังการปรับโมเดลด้วย SMOTE

คลาส	**จำนวนก่อนใช้ SMOTE**	**หลังใช้ SMOTE**	**ข้อมูลรวมทั้งหมด**
Requirements Engineer	41,450	74,720	33,270
Software Designer	26,890	74,720	47,830
Developer	73,999	74,720	721
Software Tester	70,893	74,720	3,827
Maintenance Engineer	31,825	74,720	42,895
Project Manager	1,868	74,720	72,852
System Analyst	90,529	90,529	0
รวม	**337,454**	**538,849**	**201,395**

ในการวิเคราะห์และจำแนกตำแหน่งงานจากทักษะ (Hard Skill และ Soft Skill) ที่ได้จากการดึงข้อมูลประกาศรับสมัครงาน ผู้วิจัยได้ใช้ข้อมูลที่ผ่านการแปลงให้อยู่ในรูปแบบ One-hot encoding และ TF-IDF vectorization จากนั้นนำไปฝึกโมเดลด้วยเทคนิค Machine Learning และ Deep Learning ทั้งหมด 4 แบบ ได้แก่

5.1) Random Forest

โมเดลตั้งค่าจำนวนต้นไม้ (n_estimators) เท่ากับ 100 โดยไม่มีการจำกัดความลึกของต้นไม้ (max_depth=None) เพื่อให้สามารถเรียนรู้โครงสร้างข้อมูลได้อย่างเต็มที่ ทั้งนี้ใช้ random_state=42 เพื่อให้ได้ผลลัพธ์ที่ reproducible และใช้ n_jobs=-1 เพื่อเร่งประสิทธิภาพด้วยการประมวลผล

5.2) XGBoost

ผู้วิจัยกำหนดจำนวนต้นไม้ไว้ที่ 200 ต้น (n_estimators=200) และใช้ความลึกของต้นไม้ (max_depth) เท่ากับ 6 พร้อม learning rate = 0.1 ซึ่งช่วยควบคุมอัตราการเรียนรู้ไม่ให้เร็วเกินไป รวมถึงใช้ subsample และ colsample_bytree เท่ากับ 0.8 เพื่อสุ่มตัวอย่างข้อมูลและฟีเจอร์บางส่วนในการฝึกแต่ละรอบ ช่วยลด overfitting

สำหรับการวิเคราะห์โมเดล การวิเคราะห์ในส่วนนี้ยังได้เพิ่มเติมส่วนของการวิเคราะห์โดยใช้ความถี่ของทักษะร่วมด้วยเพื่อให้ข้อมูลที่วิเคราะห์นั้นสามารถวิเคราะห์ได้ถูกต้องมากขึ้น Random Forest และ XGBoost มีการวิเคราะห์ด้วย TF-IDF vectorization ซึ่งเป็นการนำความถี่มาใช้ในการสร้างโมเดล เพื่อให้โมเดลเห็นว่า sub-skill ใดสำคัญกว่าจากข้อมูลที่ปรากฏ

5.3) LSTM (Long Short-Term Memory)

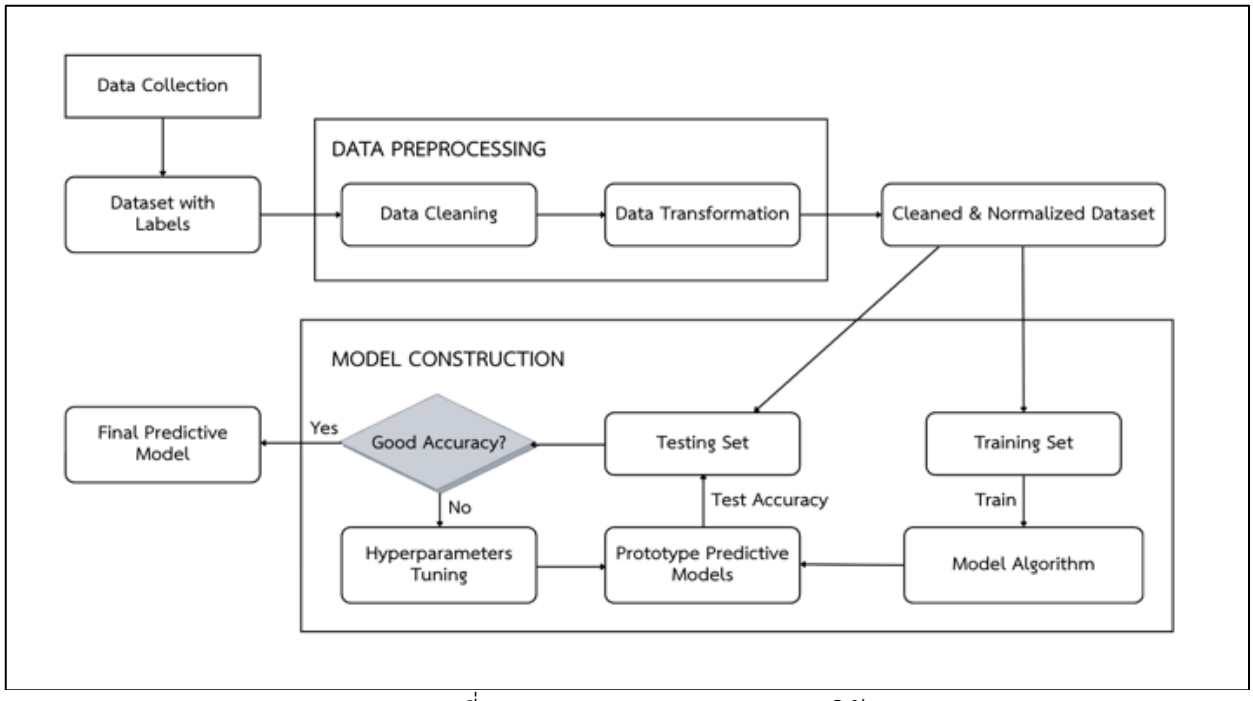
โมเดลมีโครงสร้างแบบ Bidirectional LSTM 2 ชั้น โดยชั้นแรกมีหน่วยความจำ 100 units และชั้นที่สอง 50 units ใช้ dropout และ recurrent dropout อย่างเหมาะสม และฝึกโมเดลด้วยจำนวนรอบ (epochs) เท่ากับ 100 โดยใช้ optimizer แบบ Adam พร้อม regularization เพื่อป้องกัน overfitting และใช้ early stopping ร่วมกับ class weighting เพื่อปรับสมดุลระหว่างคลาส

5.4) Naïve Bayes

ผู้วิจัยใช้อัลกอริทึม Multinomial Naïve Bayes ซึ่งเหมาะสำหรับข้อมูลเชิงข้อความที่อยู่ในรูป TF-IDF และเหมาะกับการจำแนกประเภทที่มีลักษณะเป็นการนับถี่ของคำหรือคุณลักษณะไม่ต่อเนื่อง

6) เปรียบเทียบตัวแบบทำนายสำหรับข้อมูลประกาศรับสมัครงาน ใช้กระบวนการ Train-Test Split (80:20) ได้แก่ Accuracy, Precision, Recall และ F1-score

โดยกระบวนการดังกล่าวเป็นไปตาม ภาพที่ 3-1



ภาพที่ 3-1 การออกแบบกระบวนการวิจัย

บทที่ 4
ผลการดำเนินการวิจัย

การวิจัยครั้งนี้มีวัตถุประสงค์ 1) เพื่อศึกษาข้อมูลทักษะที่ต้องการขององค์กรซอฟต์แวร์ที่ประกาศรับตำแหน่งงานด้านซอฟต์แวร์จากเว็บไซต์รับสมัครงานชั้นนำในประเทศไทย 2) เพื่อนำเสนอชุดของทักษะตามความต้องการในแต่ละตำแหน่งงานของวิศวกรซอฟต์แวร์ไทย และ 3) เพื่อสร้างตัวแบบสำหรับประเมินตำแหน่งงานที่เหมาะกับสมรรถนะในตำแหน่งงานที่กำหนด ผลการวิเคราะห์ค่าสถิติพื้นฐานของข้อมูลในการวิจัย ผู้วิจัยจึงนำเสนอผลการวิเคราะห์ดังนี้

4.1 ผลการวิเคราะห์ความถี่และร้อยละของสมรรถนะที่ได้จากแหล่งข้อมูลประกาศรับสมัครงาน

1) ข้อมูลประกาศรับสมัครงานทักษะด้านเทคนิคและด้านบุคคลจากประกาศรับสมัครงาน

ในส่วนนี้ผู้วิจัยผลการค้นพบข้อมูลที่เกิดจากประกาศรับสมัครงานในตำแหน่งที่เกี่ยวข้องกับวิศวกรรมซอฟต์แวร์ โดยได้รวบรวมข้อมูลชุดประกาศแบบสุ่มมาจำนวนทั้งหมด 1,132,871 ประกาศงาน จากเว็บไซต์ที่มีการอนุญาตให้ใช้ข้อมูลแบบสาธารณะอันได้แก่ LinkedIn และ JobThai.com โดยเก็บข้อมูลตั้งแต่เดือนตุลาคม พ.ศ 2565 ถึง ธันวาคม พ.ศ 2567 โดยข้อมูลดังกล่าวมีการเก็บย้อนหลังโดยดูจากวันที่ประกาศ 2564 ถึง 2567 และเป็นประกาศรับสมัครงานภาษาอังกฤษเท่านั้น

จากการเก็บรวบรวมข้อมูลดังกล่าวเมื่อผ่านเข้าสู่กระบวนการทำความสะอาดข้อมูล (Data cleaning) ตรวจสอบโดยผู้เชี่ยวชาญเพื่อยืนยันความถูกต้องของข้อมูลพร้อมทั้งจัดรูปแบบข้อมูลเพื่อให้ดำเนินการวิเคราะห์ข้อมูลแล้วจึงทำให้ข้อมูลทั้งหมดจำนวนประกาศทั้งหมดหลังจากทำความสะอาดข้อมูลแล้วเท่ากับ 337,454 รายการ แสดงดังตารางที่ 4-1 แสดงจำนวนประกาศรับสมัครงานตามตำแหน่งงานในอุตสาหกรรมซอฟต์แวร์ของประเทศไทย จำนวนคำสำคัญทั้ง Hard skill และ Soft skill ที่ปรากฏในแต่ละตำแหน่งงานจากข้อมูลพบว่า ตำแหน่งที่มีประกาศมากที่สุดคือ System Analyst จำนวน 90,529 รายการ รองลงมาคือ Developer และ Software Tester ขณะที่ตำแหน่งที่มีจำนวนน้อยที่สุดคือ Project Manager ในด้าน Hard skill ตำแหน่ง Developer มีจำนวนคำสำคัญสูงที่สุดถึง 100,175 คำ ซึ่งสอดคล้องกับบทบาทที่เน้นทักษะด้านเทคนิค ส่วนด้าน Soft skill พบว่าตำแหน่ง System Analyst มีจำนวนคำสำคัญสูงที่สุดที่ 69,392 คำ รองลงมาคือ Software Tester และ Developer ซึ่งสะท้อนถึงความสำคัญของทักษะด้านการสื่อสาร การวิเคราะห์ และการทำงานร่วมกับผู้อื่น

ตารางที่ 4-1 จำนวนประกาศรับสมัครงานจำแนกตามตำแหน่งงาน

ลำดับ	ตำแหน่งงาน	จำนวนประกาศ	จำนวนคำสำคัญ Hard skill	จำนวนคำสำคัญ Soft skill
1.	Requirements Engineer	41,450	27,811	35,781
2.	Software Designer	26,890	30,415	18,371
3.	Developer	73,999	100,175	42,405
4.	Software Tester	70,893	65,535	52,606
5.	Maintenance Engineer	31,825	22,370	25,556
6.	Project Manager	1,868	2,388	11,768
7.	System Analyst	90,529	76,561	69,392
รวม		**337,454**	**325,255**	**255,879**

2) ข้อมูลประกาศรับสมัครงานทักษะด้านเทคนิค

จากการข้อมูลประกาศรับสมัครงานทั้งหมด 337,454 รายการ พบทักษะด้านเทคนิค จำนวน 321,921 รายการซึ่งแบ่งผู้วิจัยได้จำแนกกลุ่มทักษะด้านเทคนิคออกเป็น 10 กลุ่ม ในตารางที่ 4-2

ตารางที่ 4-2 จำนวนและร้อยละทักษะทั้งหมดประกาศรับสมัครงานจำแนกทักษะเชิงเทคนิค

#	ทักษะ	จำนวน	ร้อยละ
1.	Programming Languages	113,966	35.40
2.	Frameworks & Libraries	47,963	14.90
3.	Cloud & Infrastructure	46,180	14.35
4.	Database & Data Technologies	32,707	10.16
5.	Software Architecture & Design	29,398	9.13
6.	Development Tools	28,820	8.95
7.	AI/ML/Data Science	8,587	2.67
8.	Testing Tools & Automation	7,443	2.31
9.	Security Tools & Techniques	6,312	1.96
10.	Mobile Development	545	0.17
	รวม	**321,921**	**100.00%**

จากตารางที่ 4-2 แสดงให้เห็นว่ากลุ่มทักษะที่มีการกล่าวถึงมากที่สุดคือ Programming Languages คิดเป็นร้อยละ 35.40 ของทักษะทั้งหมด รองลงมาคือกลุ่มทักษะ Frameworks & Libraries และ Cloud &

Infrastructure ซึ่งมีสัดส่วนร้อยละ 14.90 และ 14.35 ตามลำดับ กลุ่มทักษะ Database & Data Technologies กลุ่มทักษะ Software Architecture & Design และกลุ่มทักษะ Development Tools มีสัดส่วนที่ใกล้เคียงกัน คือ ร้อยละ 10.16, 9.13 และ 8.95 และสำหรับกลุ่ม ทักษะมีสัดส่วนน้อย คือ กลุ่มทักษะ AI/ML/Data Science, Testing Tools & Automation และ Security Tools & Techniques มีสัดส่วนร้อยละ 2.67, 2.31 และ 1.96 ตามลำดับ และ กลุ่มที่มีสัดส่วนน้อยที่สุดคือ Mobile Development คิดเป็นเพียงร้อยละ 0.17

2.1) ผลการวิเคราะห์ข้อมูลทักษะด้านเทคนิคภาษาโปรแกรม

การวิเคราะห์ทักษะด้านเทคนิคในกลุ่มภาษาโปรแกรมจากประกาศรับสมัครงานจำนวน 337,454 รายการ พบจำนวน 113,966 รายการ ผู้วิจัยเสนอทักษะที่พบจากประกาศรับสมัครงานที่อยู่ในกลุ่ม 90% - 95% และกลุ่มที่พบน้อยที่สุดไว้ในกลุ่มอื่นๆ โดยมีรายละเอียดจำนวนและสัดส่วนของแต่ละภาษาโปรแกรมที่ใช้จริงในภาคอุตสาหกรรมซอฟต์แวร์ตารางที่ 4-3

ตารางที่ 4-3 จำนวนและร้อยละของทักษะด้านภาษาโปรแกรม

#	**ทักษะด้านภาษาโปรแกรม**	**จำนวน**	**ร้อยละ**
1.	Java	19,449	17.07
2.	SQL	18,398	16.14
3.	Python	13,769	12.08
4.	JavaScript	12,912	11.33
5.	C++	11,714	10.28
6.	Kotlin	7,100	6.23
7.	Swift	6,600	5.79
8.	Typescript	3,708	3.25
9.	Go	3,587	3.15
10.	Php	3,194	2.80
11.	R	2,517	2.21
12.	Objective-c	2,404	2.11
13.	Ruby	1,129	0.99
14.	Scala	1,062	0.93
15.	Powershell	1,050	0.92
16.	Other (12 คำ)	5,373	4.71
รวม		**113,966**	**100.00**

ผลการวิเคราะห์แสดงให้เห็นว่า ภาษา Java เป็นภาษาที่ปรากฏมากที่สุด คิดเป็น 17.07% ของทักษะด้านภาษาโปรแกรมทั้งหมด รองลงมาคือ SQL (16.14%), Python (12.08%), JavaScript (11.33%) และ C++ (10.28%) ซึ่งรวมกันคิดเป็นสัดส่วนถึงกว่า 66% ของประกาศงานทั้งหมดในกลุ่มนี้

สำหรับ Kotlin (6.23%), Swift (5.79%), และ Go (3.15%) เป็นกลุ่มภาษาใหม่และเฉพาะทางสะท้อนให้เห็นถึงแนวโน้มของการพัฒนาแอปพลิเคชันบนแพลตฟอร์ม Android/iOS และระบบที่ต้องการความเสถียรแต่มีความยืดหยุ่นสูง สำหรับภาษาอื่น ๆ เช่น Objective-C, R, Scala, และ Powershell แม้จะมีสัดส่วนน้อยกว่า 3% กลุ่ม Other (12 คำ) คิดเป็น 4.71% ของทั้งหมด แสดงให้เห็นถึงความหลากหลายของภาษาที่ใช้งานจริงในอุตสาหกรรม

2.2) ผลการวิเคราะห์ข้อมูลทักษะด้านเทคนิค Frameworks & Libraries

การวิเคราะห์ทักษะด้านเทคนิค Frameworks & Libraries จำนวน 113,966 รายการ จากประกาศรับสมัครงานจำนวน 337,454 รายการ ผู้วิจัยเสนอทักษะที่พบจากประกาศรับสมัครงานที่อยู่ในสัดส่วน 90% - 95% และกลุ่มที่พบน้อยที่สุดไว้ในกลุ่มอื่นๆ โดยมีรายละเอียดจำนวนและสัดส่วนของแต่ละทักษะด้าน Frameworks & Libraries ที่ใช้จริงในภาคอุตสาหกรรมซอฟต์แวร์ตารางที่ 4-4

ตารางที่ 4-4 จำนวนและร้อยละของทักษะด้าน Frameworks & Libraries

#	Frameworks & Libraries	จำนวน	ร้อยละ
1.	React	9,708	20.24
2.	React native	8,820	18.39
3.	Flutter	6,220	12.97
4.	Spring	5,735	11.96
5.	Angular	4,291	8.95
6.	Spring boot	3,182	6.63
7.	Vue.js	2,284	4.76
8.	Jquery	2,155	4.49
9.	Bootstrap	1,159	2.42
10.	Laravel	827	1.72
11.	Angularjs	817	1.70
12.	Next.js	633	1.32
13.	Other (15 ทักษะ)	2,132	4.45
รวม		**47,963**	**100.00**

ผลการวิเคราะห์พบว่า React เป็น Framework ที่ได้รับความนิยมสูงสุดในประกาศรับสมัครงาน โดยมีการระบุถึงมากถึง 9,708 รายการ คิดเป็นร้อยละ 20.24 ของทักษะในกลุ่มนี้ รองลงมาคือ React Native (18.39%) และ Flutter (12.97%) ให้เห็นถึงความนิยมในการพัฒนา Web Application และ Mobile Application ด้วย Framework สมัยใหม่ที่รองรับการพัฒนาแบบ Cross-platform ต่อมา Spring (11.96%) และ Spring Boot (6.63%) ซึ่งเป็น Framework ฝั่ง Java ก็ยังคงได้รับความนิยมอย่างต่อเนื่องในองค์กรขนาดใหญ่และงานระดับ Enterprise และ กลุ่ม Angular (8.95%), Vue.js (4.76%) และ JQuery (4.49%) เป็นตัวแทนของ Frontend Framework ที่ยังคงมีการใช้อยู่ในอุตสาหกรรมซอฟต์แวร์ ต่อมา Bootstrap (2.42%) ยังคงปรากฏในหลายประกาศรับสมัครงาน สะท้อนให้เห็นถึงความนิยมของ CSS Framework สำหรับการออกแบบ UI นอกจากนี้ยังมี Framework ที่เฉพาะเจาะจงกับการพัฒนา Web เช่น Laravel (1.72%) สำหรับ PHP และ Next.js (1.32%) สำหรับ React ส่วนกลุ่มอื่นๆ อีก 15 ทักษะ คิดเป็นสัดส่วนร้อยละ 4.45 ซึ่งรวม Framework ที่ไม่ปรากฏบ่อยแต่ละรายการจะมีสัดส่วนค่อนข้างต่ำ

2.3) ผลการวิเคราะห์ข้อมูลทักษะด้านเทคนิค Frameworks & Libraries

การวิเคราะห์ทักษะด้านเทคนิค Cloud & Infrastructure จำนวน 46,180 รายการ จากประกาศรับสมัครงานจำนวน 337,454 รายการ ผู้วิจัยเสนอทักษะที่พบจากประกาศรับสมัครงานที่อยู่ในสัดส่วน 90% - 95% และกลุ่มที่พบน้อยที่สุดไว้ในกลุ่มอื่นๆ ดังตารางที่ 4-5

ตารางที่ 4-5 จำนวนและร้อยละของทักษะด้าน Cloud & Infrastructure

#	Cloud & Infrastructure	จำนวน	ร้อยละ
1.	AWS (Amazon Web Services)	12,699	27.50
2.	Azure	12,073	26.14
3.	Kubernetes	5,359	11.60
4.	Docker	4,943	10.70
5.	Gcp (Google Cloud Platform)	2,668	5.78
6.	Ansible	2,652	5.74
7.	VMware	2,576	5.58
8.	Terraform	2,226	4.82
9.	Other (8 ทักษะ)	984	2.13
	รวม	**46,180**	**100.00**

ผลการวิเคราะห์พบว่า AWS (Amazon Web Services) เป็นทักษะด้าน Cloud ที่ถูกระบุมากที่สุดในประกาศรับสมัครงาน โดยมีจำนวน 12,699 รายการ คิดเป็นร้อยละ 27.50 ของทักษะในกลุ่มนี้ รองลงมาคือ Microsoft Azure ซึ่งมีสัดส่วนใกล้เคียงกันที่ร้อยละ 26.14 (12,073 รายการ) สะท้อนให้เห็นว่า AWS และ

Azure ยังคงเป็นแพลตฟอร์ม Cloud ชั้นนำที่ได้รับความนิยมสูงในภาคอุตสาหกรรม โดยเฉพาะในองค์กรที่ต้องการโครงสร้างพื้นฐานแบบบริการ (Infrastructure-as-a-Service: IaaS)สำหรับ Kubernetes (11.60%) และ Docker (10.70%) เป็นเทคโนโลยีด้าน Containerization ที่ปรากฏในประกาศรับสมัครงาน GCP (Google Cloud Platform) ปรากฏในสัดส่วนร้อยละ 5.78 ซึ่งน้อยกว่า AWS และ Azure แต่ก็ยังเป็นที่ต้องการในองค์กร เทคโนโลยีสนับสนุนด้าน Automation และ Configuration เช่น Ansible (5.74%) และ Terraform (4.82%) รวมถึง VMware (5.58%) ก็ยังคงมีบทบาทในระบบที่ผสมผสานการจัดการทรัพยากรแบบ On-Premise และ Cloud-Based

สำหรับ หมวด Other (8 ทักษะ) คิดเป็นสัดส่วนร้อยละ 2.13 ของทั้งหมดปรากฏในปริมาณน้อยและใช้ในบางกลุ่มธุรกิจเท่านั้น

2.4) ผลการวิเคราะห์ข้อมูลทักษะด้านเทคนิค Database & Data Technologies

การวิเคราะห์ทักษะด้านเทคนิค Database & Data Technologies จำนวน 32,707รายการ จากประกาศรับสมัครงานจำนวน 337,454 รายการ ผู้วิจัยเสนอทักษะที่พบจากประกาศรับสมัครงานที่อยู่ในสัดส่วน 90% - 95% และกลุ่มที่พบน้อยที่สุดไว้ในกลุ่มอื่นๆ ดังตารางที่ 4-6

ตารางที่ 4-6 จำนวนและร้อยละของทักษะด้าน Database & Data Technologies

#	Database & Data Technologies	จำนวน	ร้อยละ
1.	Oracle	8,548	26.14
2.	Redis	6,724	20.56
3.	Mysql	5,072	15.51
4.	Sql server	3,383	10.34
5.	Mongodb	2,778	8.49
6.	Postgresql	2,059	6.30
7.	Apache spark	1,156	3.53
8.	Elasticsearch	756	2.31
9.	Databricks	630	1.93
10.	Other (12 ทักษะ)	1,601	4.89
	รวม	**32,707**	**100.00**

ผลการวิเคราะห์พบว่า Oracle เป็นระบบฐานข้อมูลที่ได้รับความนิยมมากที่สุด โดยปรากฏในประกาศรับสมัครงานจำนวน 8,548 รายการ คิดเป็นร้อยละ 26.14 ของทักษะในกลุ่มนี้ รองลงมาคือ Redis (20.56%) และ MySQL (15.51%) ซึ่งสะท้อนให้เห็นถึงการใช้ระบบฐานข้อมูลที่มีทั้งแบบเชิงพาณิชย์ (Oracle) และแบบโอเพนซอร์ส (Redis, MySQL) รองลงมา SQL Server (10.34%) ยังคงเป็นที่นิยมในกลุ่มองค์กรที่พัฒนาแอป

พลิเคชันด้วยเทคโนโลยีของ Microsoft ขณะที่ MongoDB (8.49%) และ PostgreSQL (6.30%) แสดงให้เห็นถึงความนิยมของฐานข้อมูลแบบ NoSQL และฐานข้อมูลเชิงสัมพันธ์แบบโอเพนซอร์สในระบบที่ต้องการความยืดหยุ่น ส่วน Apache Spark (3.53%), Elasticsearch (2.31%) และ Databricks (1.93%) แม้จะมีสัดส่วนน้อยกว่า แต่ก็สะท้อนให้เห็นถึงการใช้งานในองค์กรที่มุ่งเน้นเทคโนโลยีที่เกี่ยวข้องกับการประมวลผลข้อมูลขนาดใหญ่ และ กลุ่มอื่นๆ อีก 12 ทักษะ คิดเป็นร้อยละ 4.89 ของทั้งหมด ซึ่งเป็นเทคโนโลยีฐานข้อมูลเฉพาะทางที่ใช้น้อย เช่น Amazon Redshift, Neo4j, Memcached, Google Bigquery และอื่น ๆ

2.5) ผลการวิเคราะห์ข้อมูลทักษะด้านเทคนิค Software Architecture & Design

การวิเคราะห์ทักษะด้านเทคนิค Software Architecture & Design จำนวน 29,398 รายการ จากประกาศรับสมัครงานจำนวน 337,454 รายการ ผู้วิจัยเสนอทักษะที่พบจากประกาศรับสมัครงานที่อยู่ในสัดส่วน 90% - 95% และกลุ่มที่พบน้อยที่สุดไว้ในกลุ่มอื่นๆ ดังตารางที่ 4-7

ตารางที่ 4-7 จำนวนและร้อยละของทักษะด้าน Software Architecture & Design

#	Software Architecture & Design	จำนวน	ร้อยละ
1.	Command	10,408	35.40
2.	Strategy	6,848	23.29
3.	Microservices	4,008	13.63
4.	Design patterns	2,000	6.80
5.	MVC	1,767	6.01
6.	Solution Architecture	1,558	5.30
7.	Factory	807	2.75
8.	Bridge	612	2.08
9.	Other (11 ทักษะ)	1,390	4.73
	รวม	**29,398**	**100.00**

ผลการวิเคราะห์พบว่า ทักษะที่ปรากฏมากที่สุดคือ Command Pattern โดยมีจำนวน 10,408 รายการ คิดเป็นร้อยละ 35.40 ของทักษะในกลุ่มนี้ รองลงมาคือ Strategy Pattern (23.29%) และ Microservices Architecture (13.63%) ซึ่งสะท้อนถึงการให้ความสำคัญกับ แนวคิดด้านการออกแบบซอฟต์แวร์ที่สามารถแยกหน้าที่ความรับผิดชอบ และเพิ่มความยืดหยุ่นในการปรับเปลี่ยนพฤติกรรมของระบบ สำหรับ Design Patterns ในภาพรวม เช่น Command, Strategy, Factory, Bridge มีสัดส่วนรวมมากกว่าร้อยละ 70 ซึ่งแสดงให้เห็นว่าบริษัทต่าง ๆ ยังคงให้ความสำคัญกับพื้นฐานแนวคิดด้าน OOP (Object-Oriented Programming) และ Reusable Design Solutions เพื่อให้โค้ดสามารถบำรุงรักษาและขยายได้ในระยะยาว ทักษะ Microservices Architecture (13.63%) ปรากฏในระดับที่สูงเช่นกัน สะท้อนแนวโน้มของ

การเปลี่ยนแปลงสถาปัตยกรรมระบบจากแบบ Monolithic ไปสู่การแยกบริการย่อย (loosely coupled services) ที่สามารถพัฒนาและ deploy ได้อย่างอิสระ

นอกจากนี้ยังพบว่าทักษะเชิงโครงสร้าง เช่น MVC (6.01%) และ Solution Architecture (5.30%) ก็ปรากฏในสัดส่วนที่สำคัญ ซึ่งแสดงให้เห็นว่าองค์กรต่าง ๆ ให้ความสำคัญกับแนวคิดเชิงโครงสร้างในการจัดการกับซอฟต์แวร์ในระดับระบบและโมดูล สำหรับ กลุ่มอื่นๆ อีก 11 ทักษะ คิดเป็นร้อยละ 4.73 ของทักษะทั้งหมด ซึ่งคาดว่าจะรวมแนวคิดอื่น ๆ เช่น Singleton, Adapter, Proxy, Domain-Driven Design (DDD) และ Event-Driven Architecture ที่แม้จะไม่ปรากฏบ่อย แต่ก็มีบทบาทสำคัญในระบบเฉพาะทาง

2.6) ผลการวิเคราะห์ข้อมูลทักษะด้านเทคนิค Development Tools

การวิเคราะห์ทักษะด้านเทคนิค Development Tools จำนวน 28,820 รายการ จากประกาศรับสมัครงานจำนวน 337,454 รายการ ผู้วิจัยเสนอทักษะที่พบจากประกาศรับสมัครงานที่อยู่ในสัดส่วน 90% - 95% จำนวน 8 ทักษะ และกลุ่มที่พบน้อยที่สุดไว้ในกลุ่มอื่นๆ ดังตารางที่ 4-8

ตารางที่ 4-8 จำนวนและร้อยละของทักษะด้าน Development Tools

#	Development Tools	จำนวน	ร้อยละ
1.	Jenkins	12,664	43.94
2.	Git	5,187	18.00
3.	Jira	3,876	13.45
4.	Gitlab	1,847	6.41
5.	Confluence	1,400	4.86
6.	GitHub	1,392	4.83
7.	Bitbucket	789	2.74
8.	Gitlab CI	624	2.17
9.	Other (11 ทักษะ)	1,041	3.61
	รวม	**28,820**	**100.00**

ผลการวิเคราะห์พบว่า Jenkins เป็นเครื่องมือที่ได้รับการกล่าวถึงมากที่สุด โดยมีจำนวน 12,664 รายการ คิดเป็นร้อยละ 43.94 ของทักษะทั้งหมดในกลุ่มนี้ ซึ่งสะท้อนให้เห็นถึงความนิยมของเครื่องมือในกลุ่ม CI/CD (Continuous Integration / Continuous Deployment) ที่ช่วยให้การ Build, Test และ Deploy ซอฟต์แวร์เป็นไปโดยอัตโนมัติและต่อเนื่อง รองลงมา Git (18.00%) ซึ่งเป็นเครื่องมือเวอร์ชันคอนโทรลพื้นฐานที่ใช้ในการพัฒนาแบบทีม ส่วน Jira (13.45%) ซึ่งเป็นระบบจัดการงาน และเครื่องมืออย่าง GitLab (6.41%), GitHub (4.83%), และ Bitbucket (2.74%) ซึ่งล้วนเป็นแพลตฟอร์มการจัดการ Repository และ CI/CD ต่อมา Confluence (4.86%) ที่เน้นการจัดการความรู้ (knowledge management) และเอกสารประกอบ

ระบบ และ GitLab CI (2.17%) ซึ่งเป็นส่วนเสริมของ GitLab สำหรับการทำ CI/CD ก็พบได้ในหลายประกาศงานเช่นกัน สำหรับกลุ่มอื่นๆ อีก 11 ทักษะ คิดเป็นร้อยละ 3.61 ซึ่งเป็นเครื่องมือที่ใช้เฉพาะด้านภายในองค์กร เช่น Trello, SVN และ CircleCI เป็นต้น

2.7) ผลการวิเคราะห์ข้อมูลทักษะด้านเทคนิค AI/ML/Data Science

การวิเคราะห์ทักษะด้านเทคนิค Development Tools จำนวน 28,820 รายการ จากประกาศรับสมัครงานจำนวน 337,454 รายการ ผู้วิจัยเสนอทักษะที่พบจากประกาศรับสมัครงานที่อยู่ในสัดส่วน 90% - 95% จำนวน 9 ทักษะ และกลุ่มที่พบน้อยที่สุดไว้ในกลุ่มอื่นๆ ดังตารางที่ 4-9

ตารางที่ 4-9 จำนวนและร้อยละของทักษะด้าน AI/ML/Data Science

#	AI/ML/Data Science	จำนวน	ร้อยละ
1.	Machine learning	3,788	44.11
2.	Tableau	1,652	19.24
3.	Tensorflow	749	8.72
4.	PyTorch	647	7.53
5.	Data mining	504	5.87
6.	NLP	391	4.55
7.	Pandas	205	2.39
8.	Scikit-learn	188	2.19
9.	Keras	103	1.20
10.	Other (9 ทักษะ)	360	4.19
	รวม	**8,587**	**100.00**

ผลการวิเคราะห์พบว่า ทักษะที่ปรากฏมากที่สุดคือ Machine Learning จำนวน 3,788 รายการ คิดเป็นร้อยละ 44.11 ของทักษะกลุ่มนี้ ซึ่งสะท้อนให้เห็นว่าภาคอุตสาหกรรมยังคงให้ความสำคัญกับแนวคิดพื้นฐานของการเรียนรู้ของเครื่องมากกว่าการใช้งานเครื่องมือเฉพาะทาง รองลงมาคือ Tableau (19.24%) ซึ่งเป็นเครื่องมือสำหรับการแสดงผลข้อมูล (Data Visualization) ที่ใช้งานง่ายและมีความสามารถในการวิเคราะห์ข้อมูลเชิงธุรกิจ จึงได้รับความนิยมในงานด้าน Business Intelligence (BI) และ Data Analyst สำหรับเครื่องมือสำหรับพัฒนาและประมวลผลโมเดลอย่าง TensorFlow (8.72%) และ PyTorch (7.53%) ก็ปรากฏในระดับสูง สะท้อนถึงความนิยมของ Python-Based Deep Learning Frameworks ที่รองรับงานด้านภาพ เสียง และข้อความได้หลากหลาย สำหรับทักษะอื่น ๆ ที่ปรากฏในระดับกลาง ได้แก่ Data Mining (5.87%), Natural Language Processing – NLP (4.55%), และ Pandas (2.39%) ซึ่งเกี่ยวข้องกับการสกัดข้อมูล การประมวลผลข้อความ และการจัดการชุดข้อมูลตามลำดับ และ Scikit-learn (2.19%) และ Keras

(1.20%) แม้จะมีสัดส่วนน้อยกว่า แต่ยังคงมีบทบาทในระดับการใช้งานจริงในงานวิเคราะห์ข้อมูลและโมเดล machine learning แบบเบื้องต้น และสำหรับกลุ่มอื่นๆ อีก 9 ทักษะ คิดเป็นร้อยละ 4.19 ของทั้งหมด ซึ่งอาจรวมถึงเครื่องมือหรือไลบรารีเฉพาะทาง เช่น LightGBM, XGBoost, DataRobot หรือ MLFlow ที่ใช้ในวงจำกัดหรือเฉพาะบางองค์กร

2.8) ผลการวิเคราะห์ข้อมูลทักษะด้านเทคนิค Testing Tools & Automation

การวิเคราะห์ทักษะด้านเทคนิค Testing Tools & Automation จำนวน 7,433 รายการ จากประกาศรับสมัครงานจำนวน 337,454 รายการ ผู้วิจัยเสนอทักษะที่พบจากประกาศรับสมัครงานที่อยู่ในสัดส่วน 90% - 95% จำนวน 12 ทักษะ และกลุ่มที่พบน้อยที่สุดไว้ในกลุ่มอื่นๆ ดังตารางที่ 4-10

ตารางที่ 4-10 จำนวนและร้อยละของทักษะด้าน Testing Tools & Automation

#	Testing Tools & Automation	จำนวน	ร้อยละ
1.	Selenium	1,748	23.49
2.	Jmeter	899	12.08
3.	Postman	839	11.27
4.	Cucumber	657	8.83
5.	Appium	609	8.18
6.	Junit	583	7.83
7.	Cypress	490	6.58
8.	Robot framework	488	6.56
9.	Playwright	238	3.20
10.	Loadrunner	201	2.70
11.	Soapui	182	2.45
12.	Katalon	151	2.03
13.	Other (5 ทักษะ)	358	4.81
	รวม	7,443	100.00

ผลการวิเคราะห์พบว่า Selenium เป็นเครื่องมือที่ได้รับความนิยมมากที่สุด โดยปรากฏใน 1,748 รายการ คิดเป็นร้อยละ 23.49 ของกลุ่มนี้ ซึ่งสะท้อนว่า Selenium ยังคงเป็นเครื่องมืออัตโนมัติสำหรับการทดสอบ UI ที่ได้รับความนิยมสูงสุดในอุตสาหกรรมซอฟต์แวร์ เครื่องมือด้าน Performance Testing อย่าง JMeter (12.08%) และ Postman (11.27%) ที่ใช้สำหรับ API Testing รองลงมาเครื่องมือทดสอบแบบ BDD (Behavior-Driven Development) เช่น Cucumber (8.83%) และ Robot Framework (6.56%) ก็ยังคงถูกกล่าวถึงอย่างแพร่หลาย โดยใช้ร่วมกับการเขียน Test Case ที่อธิบายเป็นภาษาธรรมชาติ นอกจากนี้ ยังพบเครื่องมือเฉพาะทางอื่น ๆ เช่น Appium (8.18%) สำหรับทดสอบ Mobile App, JUnit (7.83%) สำหรับ

Unit Testing บน Java และ Cypress (6.58%) ที่เน้นการทดสอบ front-end web ด้วย JavaScript อีกทั้งยังพบกลุ่มเครื่องมือที่มีการใช้งานเฉพาะ เช่น Playwright, LoadRunner, SoapUI และ Katalon ปรากฏในสัดส่วนรวมกันราว 10% แสดงให้เห็นถึงความหลากหลายของเครื่องมือที่เลือกใช้ตามลักษณะงานและเทคโนโลยีของระบบ และกลุ่มอื่นๆ อีก 5 ทักษะ คิดเป็นร้อยละ 4.81 ซึ่งอาจรวมถึงเครื่องมือที่ใช้น้อย เช่น FitNesse, TestNG, BlazeMeter หรือ NeoLoad

2.9) ผลการวิเคราะห์ข้อมูลทักษะด้านเทคนิค Security Tools & Techniques

การวิเคราะห์ทักษะด้านเทคนิค Security Tools & Techniques จำนวน 6,312 รายการ จากประกาศรับสมัครงานจำนวน 337,454 รายการ ผู้วิจัยเสนอทักษะที่พบจากประกาศรับสมัครงานที่อยู่ในสัดส่วน 90% - 95% จำนวน 12 ทักษะ และกลุ่มที่พบน้อยที่สุดไว้ในกลุ่มอื่นๆ ดังตารางที่ 4-11

ตารางที่ 4-11 จำนวนและร้อยละของทักษะด้าน Security Tools & Techniques

#	Security Tools & Techniques	จำนวน	ร้อยละ
1.	Firewalls	1,648	26.11
2.	Penetration Testing	816	12.93
3.	Encryption	656	10.39
4.	Security Testing	654	10.36
5.	Siem	615	9.74
6.	Splunk	434	6.88
7.	Oauth	372	5.89
8.	Owasp	245	3.88
9.	Nessus	163	2.58
10.	Ips/Ids	133	2.11
11.	Burp Suite	119	1.89
12.	Data Loss Prevention	116	1.84
13.	Other (8 ทักษะ)	341	5.40
	รวม	6,312	100.00

ผลการวิเคราะห์พบว่า Firewalls เป็นทักษะด้านความปลอดภัยที่ได้รับการกล่าวถึงมากที่สุด คิดเป็น 26.11% ของทักษะในกลุ่มนี้ทั้งหมด รองลงมา Penetration testing (12.93%) และ Encryption (10.39%) เป็นทักษะเชิงรุกและเชิงป้องกันที่องค์กรให้ความสำคัญ โดยเฉพาะในระบบที่ต้องการการประเมินความเสี่ยงและการป้องกันข้อมูลสำคัญ สำหรับ Security testing, SIEM และ Splunk มีสัดส่วนระหว่าง 6–10% ซึ่งชี้ให้เห็นถึงความต้องการทักษะด้านการตรวจสอบและวิเคราะห์เหตุการณ์ด้านความปลอดภัยแบบ Real-Time

โดยเฉพาะในองค์กรขนาดกลางถึงใหญ่ นอกจากนี้ยังมีทักษะที่เกี่ยวข้องกับการจัดการสิทธิ์และช่องโหว่ เช่น OAuth, OWASP, Nessus, IPS/IDS, และ Burp Suite ซึ่งสะท้อนให้เห็นถึงความหลากหลายของเครื่องมือที่ใช้ในกระบวนการ secure SDLC และกลุ่ม อื่นๆ อีก 8 ทักษะ คิดเป็นร้อยละ 5.40 ซึ่งเป็นเทคนิคและเครื่องมือเฉพาะ เช่น Metasploit, AppScan, PKI, หรือ Security Assertion Markup Language (SAML) ที่ถูกใช้งานในกลุ่มองค์กรเฉพาะด้านหรือระบบที่มีมาตรฐานสูง

2.10) ผลการวิเคราะห์ข้อมูลทักษะด้านเทคนิค Mobile Development

การวิเคราะห์ทักษะด้านเทคนิค Mobile Development จำนวน 545 รายการ จากประกาศรับสมัครงานจำนวน 337,454 รายการ ซึ่งเป็นหมวดที่มีสัดส่วนน้อยที่สุดเมื่อเทียบกับกลุ่มทักษะเชิงเทคนิคอื่น จำนวน 6 ทักษะ ดังตารางที่ 4-12

ตารางที่ 4-12 จำนวนและร้อยละของทักษะด้าน Mobile Development

#	Mobile Development	จำนวน	ร้อยละ
1.	Xamarin	185	33.94
2.	Android sdk	154	28.26
3.	Ionic	112	20.55
4.	Cordova	70	12.84
5.	Java (android)	12	2.20
6.	Phonegap	12	2.20
	รวม	**545**	**100.00**

ผลการวิเคราะห์พบว่า Xamarin เป็นทักษะที่พบมากที่สุดในกลุ่มนี้ โดยมีจำนวน 185 รายการ คิดเป็นร้อยละ 33.94 รองลงมาคือ Android SDK (28.26%) และ Ionic (20.55%) ซึ่งล้วนเป็นแพลตฟอร์มที่รองรับการพัฒนาแบบ Cross-Platform หรือ Native Android โดยเฉพาะ รองลงมาCordova (12.84%) และ PhoneGap (2.20%) ซึ่งเป็นเทคโนโลยีแบบ Hybrid Application ก็ยังคงปรากฏอยู่ในประกาศรับสมัครแม้ว่าจะได้รับความนิยมน้อยลงเมื่อเทียบกับ Framework ที่ใหม่กว่า

รองลงมา ทักษะ Java (Android) ปรากฏในระดับต่ำ (2.20%) อาจเป็นเพราะปัจจุบันนักพัฒนามีแนวโน้มเปลี่ยนไปใช้ Kotlin หรือ Cross-Platform Tools แทน แม้จำนวนรวมของทักษะในกลุ่มนี้จะน้อยเมื่อเทียบกับกลุ่มอื่น แต่ก็แสดงให้เห็นว่า ตลาดยังคงต้องการนักพัฒนา Mobile Application โดยเฉพาะที่มีความเชี่ยวชาญในเทคโนโลยีที่สามารถทำงานข้ามแพลตฟอร์มได้

3) ข้อมูลประกาศรับสมัครงานทักษะด้านบุคคล

เมื่อพิจารณาข้อมูลประกาศรับสมัครงานที่ระบุทักษะเชิงบุคคล (Soft Skills) ทั้งหมดจำนวน 255,879 รายการ ได้พบทักษะเชิงบุคคลที่ปรากฏบ่อยที่สุดที่ผู้วิจัยเสนอไว้ในสัดส่วนประมาณ 90% จำนวน 12 ทักษะ และทักษะอื่นๆ อีก 10% โดยแสดงรายละเอียดจำนวนครั้งและร้อยละของแต่ละทักษะในตารางที่ 4-13

ตารางที่ 4-13 จำนวนและร้อยละทักษะทั้งหมดประกาศรับสมัครงานจำแนกทักษะด้านบุคคล

#	ทักษะเชิงบุคคล	จำนวน	ร้อยละ
1.	Management	63,744	24.91
2.	Communication	42,956	16.79
3.	Collaborate	38,118	14.90
4.	Problem-solving	23,624	9.23
5.	Analytical	19,205	7.51
6.	Planning	12,307	4.81
7.	Leadership	8,609	3.36
8.	Responsibility	6,262	2.45
9.	Pressure	5,399	2.11
10.	Innovation	4,156	1.62
11.	Creative	3,695	1.44
12.	Commitment	3,396	1.33
13.	Other (47 คำ)	24,408	9.54
	รวม	**255,879**	**100.00**

ผลการวิเคราะห์พบว่า ทักษะที่ปรากฏมากที่สุดคือ Management ซึ่งปรากฏจำนวน 63,744 ครั้ง คิดเป็นร้อยละ 24.91 ของทักษะทั้งหมดในกลุ่มนี้ แสดงว่าวิศวกรรมซอฟต์แวร์จำเป็นต้องมีทักษะการบริหารจัดการในงานด้านเทคโนโลยีสารสนเทศและซอฟต์แวร์ รองลงมาคือ Communication 16.79% และ Collaborate/Collaboration/Teamwork คิดเป็นประมาณ 14.9% ซึ่งเกี่ยวข้องกับความสามารถในการทำงานร่วมกับผู้อื่น และการสื่อสารที่มีประสิทธิภาพ ซึ่งเป็นองค์ประกอบสำคัญของการทำงานเป็นทีมพัฒนาซอฟต์แวร์ที่เน้นการสื่อสารระหว่างทีม ส่วนทักษะอื่นที่มีความสำคัญ ได้แก่ Analytical (7.51%) และ Problem-solving (รวม 9.23%) แสดงถึงความจำเป็นในการคิดวิเคราะห์และแก้ปัญหาอย่างเป็นระบบ

ทักษะ Planning, Leadership และ Responsibility มีสัดส่วนอยู่ในระดับกลาง (2–5%) ซึ่งเกี่ยวข้องกับการวางแผน การรับผิดชอบ และภาวะผู้นำที่ ทักษะที่ปรากฏในระดับต่ำแต่ยังคงมีความสำคัญ ได้แก่ Innovation (1.62%), Creative (1.44%) และ Commitment (1.33%) ซึ่งบ่งบอกถึงการให้ความสำคัญกับการคิดเชิงนวัตกรรม การริเริ่มสิ่งใหม่ และความมุ่งมั่น ส่วนกลุ่ม Other (47 คำ) ซึ่งรวมทักษะเชิงบุคคลที่พบ

ไม่บ่อยนัก คิดเป็นร้อยละ 9.54 ของข้อมูลทั้งหมด แสดงให้เห็นถึงความหลากหลายของคุณลักษณะบุคลิกภาพที่องค์กรให้ความสนใจในระดับเฉพาะกลุ่ม

โดยสรุปในภาพรวม การวิเคราะห์ทักษะเชิงบุคคลชี้ให้เห็นว่า ทักษะด้านการบริหารจัดการ การสื่อสาร และการทำงานร่วมกัน เป็นทักษะหลักที่ได้รับความต้องการสูงในภาคอุตสาหกรรมซอฟต์แวร์

4.2 ผลการวิเคราะห์ชุดทักษะตามความต้องการในแต่ละตำแหน่งงานของวิศวกรรมซอฟต์แวร์ไทย

ผลการวิเคราะห์ในหัวข้อนี้ ผู้วิจัยนำเสนอกลุ่มทักษะใน 2 กลุ่มหลักคือ ด้านเทคนิค และด้านบุคคลในแต่ละตำแหน่งงานจำนวน 7 ตำแหน่งงาน ได้แก่ วิศวกรความต้องการ (Requirements Engineer) นักออกแบบซอฟต์แวร์ (Software Designer) นักพัฒนาซอฟต์แวร์ (Developer Software) นักทดสอบซอฟต์แวร์ (Tester) วิศวกรซ่อมบำรุงซอฟต์แวร์ (Maintenance Engineer) ผู้จัดการโครงการซอฟต์แวร์ (Project Manager) และ นักวิเคราะห์ระบบ (System Analyst) ที่ได้กำหนดไว้ในขอบเขตการวิจัย

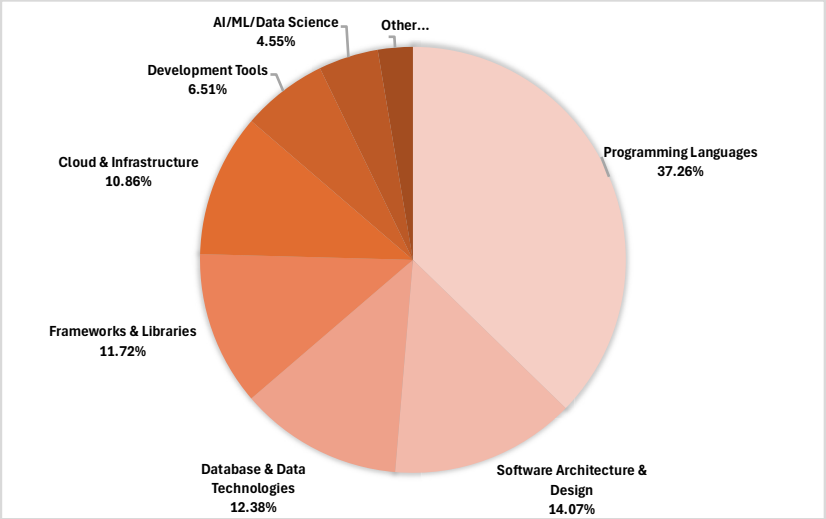
โดยในแต่ละตำแหน่งจะอธิบายถึงทักษะด้านเทคนิคในแต่ละตำแหน่งงาน และในส่วนของทักษะด้านเทคนิคดำเนินวิเคราะห์ในส่วนของทักษะย่อยจำนวน 10 กลุ่มย่อย ได้แก่ 1) Programming Languages 2) Frameworks & Libraries 3) Cloud & Infrastructure 4) Database & Data Technologies 5) Software Architecture & Design 6) Development Tools 7) AI/ML/Data Science 8) Testing Tools & Automation 9) Security Tools & Techniques และ 10) Mobile Development ใน 5 อันดับสูงสุด ส่วนทักษะด้านบุคคลนำเสนอผลสูงสุด 5 อันดับ โดยมีรายละเอียดดังนี้

1) ตำแหน่งวิศวกรความต้องการ (Requirements Engineer)

จากการรวบรวมประกาศรับสมัครงานทั้งหมดในตำแหน่งวิศวกรความต้องการ 41,450 รายการพบว่าผลการวิเคราะห์ทักษะเชิงเทคนิคที่ปรากฏในประกาศรับสมัครงานสำหรับตำแหน่งวิศวกรความต้องการ โดยพิจารณาจากกลุ่มทักษะเชิงเทคนิค 10 กลุ่ม มีทักษะด้านเทคนิค จำนวน 27,811 ทักษะ และทักษะด้านบุคคลจำนวน 35,781 ทักษะ โดยมีรายละเอียดดังนี้

1.1) ทักษะด้านเทคนิคของวิศวกรความต้องการ

ผลการวิเคราะห์ พบว่า กลุ่มทักษะที่ปรากฏสูงสุด 90% แรกอยู่ในกลุ่ม ได้แก่ Programming Languages (37.26%) รองลงมาคือกลุ่ม Software Architecture & Design (14.07%) Database & Data Technologies (12.38%) ทักษะด้าน Frameworks & Libraries (11.72%) Cloud & Infrastructure (10.86%) และ Development Tools (6.51%) ตามลำดับ ดังภาพที่ 4-1



ภาพที่ 4-1 สัดส่วนทักษะด้านเทคนิคของวิศวกรความต้องการ (Requirements Engineer)

เมื่อพิจารณาในแต่ละกลุ่มทักษะย่อยที่พบมากที่สุด 5 กลุ่มนั้นใน 5 อันดับแรก ดังนี้

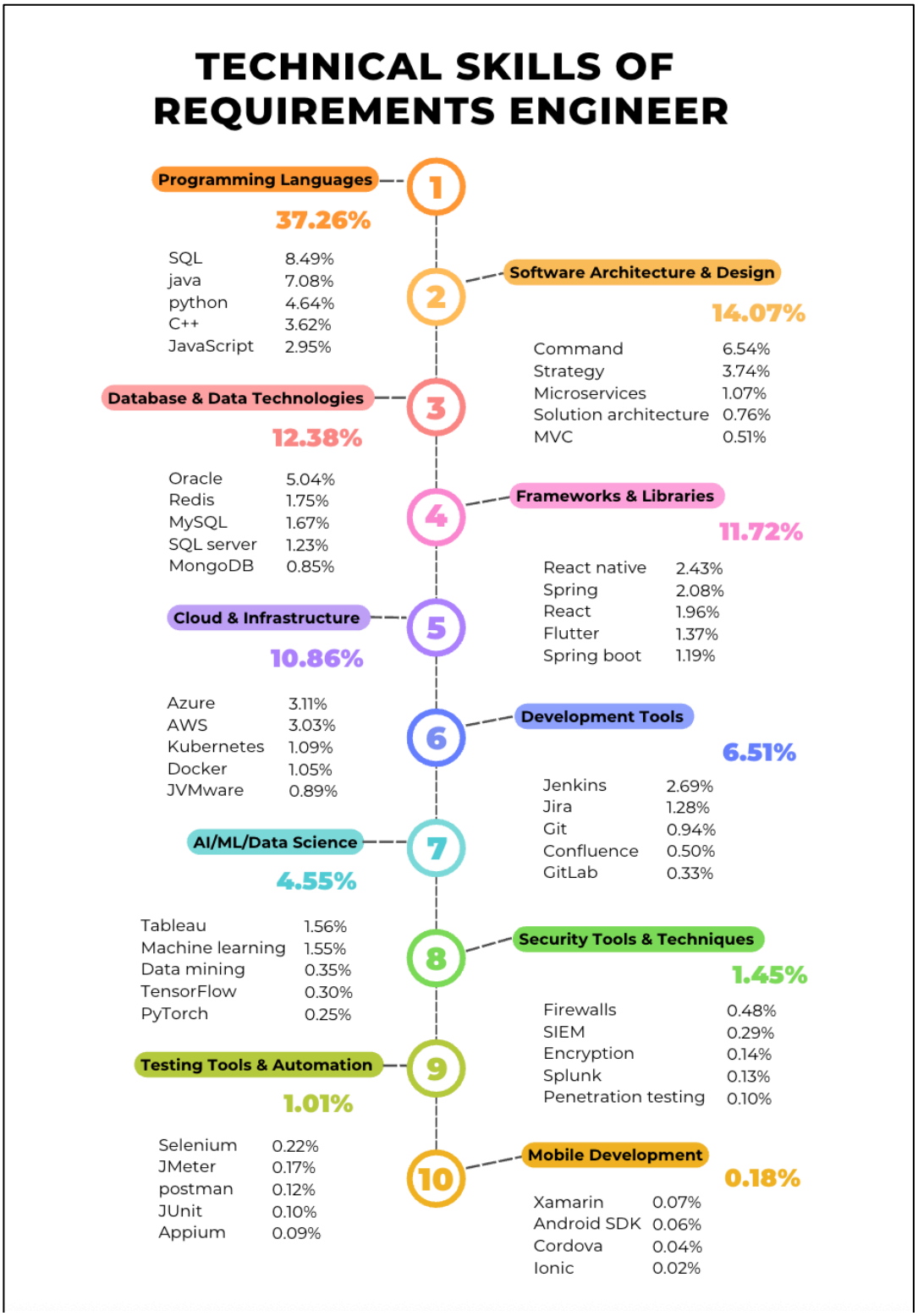
กลุ่มที่ 1: Programming Languages (37.26%) พบว่า วิศวกรความต้องการควรมีทักษะ SQL, java, python, C++ และ JavaScript

กลุ่มที่ 2: Software Architecture & Design พบว่า ที่วิศวกรความต้องการต้องเข้าใจแนวคิดการออกแบบระบบจึงต้องมีทักษะ ได้แก่ Command pattern (6.54%) และ Strategy (3.74%)

กลุ่มที่ 3: Database & Data Technologies (12.38%) ประกอบด้วยทักษะฐานข้อมูลหลากหลายที่วิศวกรความต้องการควรมี ได้แก่ Oracle, Redis, MySQL, SQL server และ MongoDBซึ่งแสดงถึงความจำเป็นในการเข้าใจโครงสร้างข้อมูลเพื่อสนับสนุนการวิเคราะห์ความต้องการที่อิงกับระบบฐานข้อมูลจริง

กลุ่มที่ 4–6: Frameworks, Cloud, Tools (รวม ~29%) กลุ่มเหล่านี้รวมทักษะด้านFramework เช่น React Native, Spring, Flutter, และเครื่องมือพัฒนาเช่น Jenkins, Jira, Git รวมถึง Cloud เช่น Azure, AWS

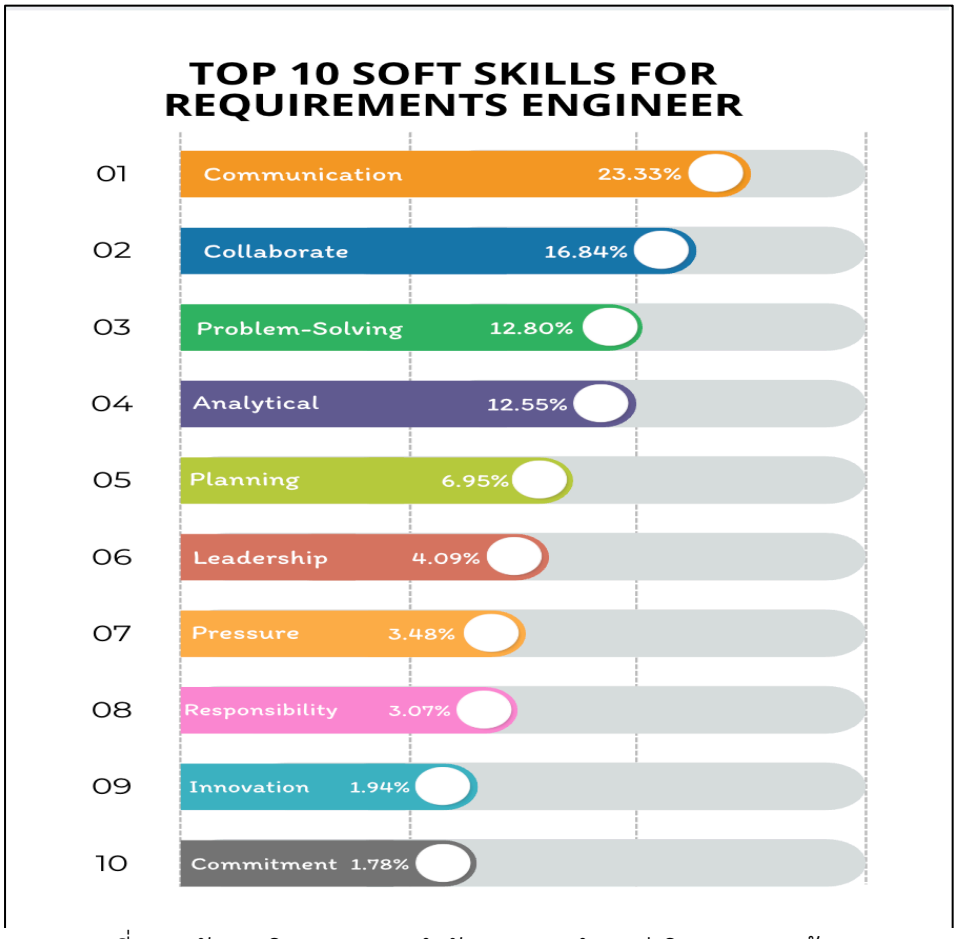
และ กลุ่มที่ 7–10: Emerging Skills (รวม ~7%) ได้แก่ทักษะ AI/ML (4.55%), Security (1.45%), Testing (1.01%) และ Mobile Development (0.18%) แม้จะมีสัดส่วนไม่มาก แต่แสดงให้เห็นว่าในบางระบบซับซ้อน วิศวกรความต้องการอาจต้องเข้าใจเทคโนโลยีใหม่หรือด้านเฉพาะ เช่น การประมวลผลข้อมูลอัจฉริยะหรือความปลอดภัยของข้อมูล ภาพที่ 4-2



ภาพที่ 4-2 ทักษะเชิงเทคนิคของตำแหน่งวิศวกรความต้องการ(Requirements Engineer)

1.2) ทักษะด้านบุคคลของวิศวกรความต้องการ

ผลการวิเคราะห์ทักษะด้านบุคคลที่เป็นความต้องการสูงที่สุด 10 ลำดับแรก ของวิศวกรความต้องการ คือ Communication (23.33%) เป็นทักษะที่สำคัญที่สุดในตำแหน่งนี้ สะท้อนบทบาทของวิศวกรความต้องการในฐานะ "ผู้เชื่อม" ระหว่างผู้ใช้งานและทีมพัฒนา ซึ่งจำเป็นต้องถ่ายทอดความต้องการได้อย่างถูกต้องครบถ้วน และเข้าใจง่าย รองลงมา Collaborate (16.84%) และ Problem-Solving (12.80%) ซึ่งเน้นความสามารถในการทำงานร่วมกับทีมข้ามสายงาน และการวิเคราะห์ทางเลือกเชิงระบบเมื่อมีความคลุมเครือหรือข้อขัดแย้งเกิดขึ้น รวมถึง Analytical (12.55%) ก็เป็นทักษะที่ปรากฏในระดับสูง สะท้อนถึงบทบาทในการทำงานกับข้อมูล ความต้องการของผู้ใช้ และกระบวนการวิเคราะห์เชิงลึก เพื่อนำมาจัดทำข้อกำหนดที่มีคุณภาพ นอกจากนี้ ทักษะอื่นที่สำคัญได้แก่ Planning (6.95%), Leadership (4.09%) และ Responsibility (3.07%) ซึ่งสะท้อนถึงความสามารถในการกำหนดลำดับความสำคัญของงาน บริหารเวลา และรับผิดชอบต่อผลลัพธ์ที่ส่งมอบ และกลุ่มทักษะ Innovation (1.94%) และ Commitment (1.78%) แสดงให้เห็นว่าแม้จะมีสัดส่วนค่อนข้างน้อย ดังภาพที่ 4-3



ภาพที่ 4-3 ทักษะเชิงบุคคล 10 ลำดับแรกของตำแหน่งวิศวกรความต้องการ

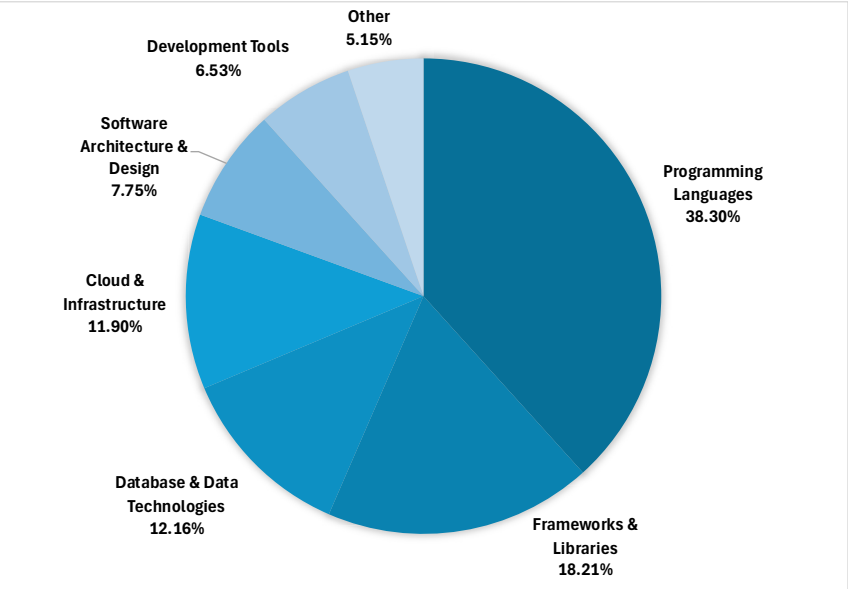
2) นักออกแบบซอฟต์แวร์ (Software Designer)

จากการรวบรวมประกาศรับสมัครงานทั้งหมดในตำแหน่งวิศวกรความต้องการ 26,890 รายการพบว่าผลการวิเคราะห์ทักษะเชิงเทคนิคที่ปรากฏในประกาศรับสมัครงานสำหรับตำแหน่งวิศวกรความต้องการ โดยพิจารณาจากกลุ่มทักษะเชิงเทคนิค 10 กลุ่ม จำนวนทักษะที่ค้นพบในกลุ่มนี้มี 30,415 รายการ และกลุ่มทักษะด้านบุคคลจำนวน 18,371 รายการ มีรายละเอียดดังต่อไปนี้

2.1) ทักษะด้านเทคนิคของนักออกแบบซอฟต์แวร์

ผลการวิเคราะห์ พบว่า กลุ่มทักษะที่ปรากฏสูงสุดประมาณ 95% แรกอยู่ในกลุ่ม ได้แก่ Programming Languages (38.30%) รองลงมาคือกลุ่ม Frameworks & Libraries (18.21%) Database & Data Technologies (12.16%) ทักษะด้าน Cloud & Infrastructure (11.90%) และ Development Tools (7.75%) Software Architecture & Design (7.75%) ตามลำดับ

ข้อมูลดังกล่าว แสดงให้เห็นว่าองค์ความรู้ด้านภาษาโปรแกรมยังคงเป็นทักษะหลักของผู้ที่ทำงานด้านการพัฒนาซอฟต์แวร์ซึ่งล้วนเป็นรากฐานในการพัฒนาโปรแกรมและระบบต่าง ๆ รวมถึงความจำเป็นที่นักพัฒนาต้องมีทักษะในการใช้เฟรมเวิร์กเพื่อเพิ่มประสิทธิภาพในการพัฒนา และความสามารถในการเชื่อมต่อและบริหารจัดการข้อมูลจากฐานข้อมูลต่าง ๆ นอกจากนี้ยังต้องมีทักษะด้าน Cloud & Infrastructure และ Software Architecture & Design ส่วนที่ขยายให้เห็นว่านักพัฒนาในยุคปัจจุบันที่ไม่เพียงแค่เขียนโค้ด แต่ยังต้องเข้าใจบริบทของการ Deploy ระบบ และการออกแบบสถาปัตยกรรม และสุดท้ายกลุ่ม Development Tools แสดงถึงความสำคัญของเครื่องมือสนับสนุนในวงจรการพัฒนาซอฟต์แวร์ ดังภาพที่ 4-4



ภาพที่ 4-4 สัดส่วนทักษะด้านเทคนิคของตำแหน่งนักออกแบบซอฟต์แวร์

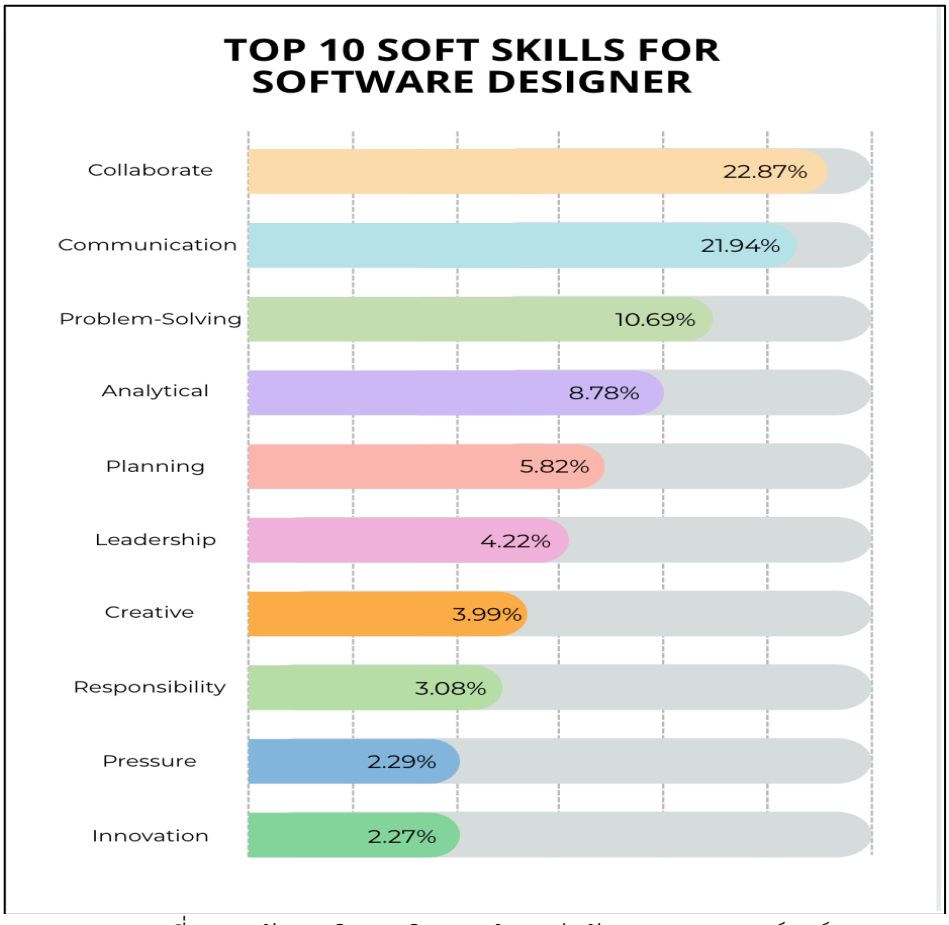
เมื่อพิจารณาทักษะย่อยในแต่ละกลุ่มทักษะที่ต้องมี 95% แรก โดยเลือกนำเสนอเฉพาะทักษะย่อยที่พบมากที่สุด 5 อันดับแรก พบว่า กลุ่มที่ 1: Programming Languages (38.30%) พบว่า นักออกแบบซอฟต์แวร์ควรมีทักษะ JavaScript, SQL, java, python และ C++ กลุ่มที่ 2: Frameworks & Libraries (18.21%) พบว่า นักออกแบบซอฟต์แวร์ควรมีทักษะ React native, React, Flutter, Angular และ Spring กลุ่มที่ 3: Database & Data Technologies (12.16%) ประกอบด้วยทักษะฐานข้อมูลหลากหลายที่นักออแบบซอฟต์แวร์ควรมี ได้แก่ Oracle, MySQL, Redis, SQL Server และ PostgreSQL กลุ่มที่ 4: Cloud & Infrastructure (11.90%) ประกอบด้วย AWS, Azure, Docker, Kubernetes และ GPC ตามลำดับ กลุ่มที่ 5: Software Architecture & Design ประกอบด้วย Strategy, Command, Microservices, Design patterns และ MVC ตามลำดับ ส่วนอีกประมาณ 5% ที่เหลือจะเป็นกลุ่มอื่นๆ ที่ยังคงเป็นทักษะที่ยังต้องการอยู่สำหรับบางหน่วยงานที่ต้องการความสามารถเฉพาะ ดังภาพที่ 4-5

ภาพที่ 4-5 ทักษะเชิงเทคนิคของตำแหน่งนักออกแบบซอฟต์แวร์

2.2) ทักษะด้านบุคคลของนักออกแบบซอฟต์แวร์

ทักษะที่พบมากที่สุดคือ Collaborate และ Communication ทั้งสองทักษะนี้คิดเป็นเกือบครึ่งหนึ่งของทั้งหมด แสดงให้เห็นว่าบทบาทของนักออกแบบซอฟต์แวร์ต้องใช้การสื่อสารอย่างและการทำงานร่วมกันระหว่างลูกค้าและทีมวิศวกร และผู้มีส่วนได้ส่วนเสียอื่น ๆ ในกระบวนการพัฒนาซอฟต์แวร์ ต่อมาคือ ทักษะ Problem-Solving และ Analytical แสดงถึงการที่นักออกแบบซอฟต์แวร์ต้องสามารถทำความเข้าใจปัญหาที่ซับซ้อน คิดวิเคราะห์โครงสร้าง และนำเสนอแนวทางแก้ไขที่เหมาะสมกับบริบทการใช้งาน

นอกจากนี้ยังพบ กลุ่มทักษะ Planning (5.82%), Leadership (4.22%), และ Creative (3.99%) บ่งบอกว่านักออกแบบไม่เพียงแต่ต้อง “คิด” และ “สื่อสาร” แต่ยังต้องสามารถจัดการกระบวนการออกแบบให้มีประสิทธิภาพ มีจุดยืนในทีม และคิดนอกกรอบเพื่อออกแบบระบบที่ตรงกับประสบการณ์ผู้ใช้ ส่วนทักษะอื่นๆ ได้แก่ Responsibility (3.08%), Pressure (2.29%), และ Innovation (2.27%) ซึ่งแม้จะมีสัดส่วนน้อยกว่า แต่ยังเป็นองค์ประกอบที่นักออกแบบซอฟต์แวร์ควรมี ดังภาพที่ ตารางที่ 4-6



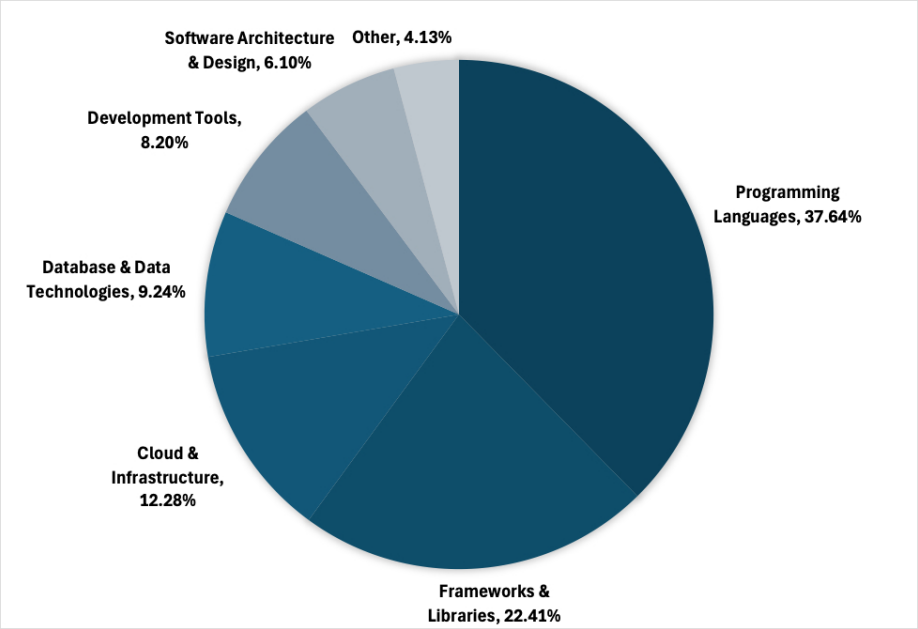
ภาพที่ 4-6 ทักษะเชิงเทคนิคของตำแหน่งนักออกแบบซอฟต์แวร์

3) นักพัฒนาซอฟต์แวร์ (Software Developer)

จากการรวบรวมประกาศรับสมัครงานทั้งหมดในตำแหน่งวิศวกรความต้องการ 73,999 รายการพบว่าผลการวิเคราะห์ทักษะเชิงเทคนิคที่ปรากฏในประกาศรับสมัครงานสำหรับตำแหน่งวิศวกรความต้องการ โดยพิจารณาจากกลุ่มทักษะเชิงเทคนิค 10 กลุ่ม มีทักษะด้านเทคนิค จำนวน 100,175 ทักษะ และทักษะด้านบุคคลจำนวน 42,405ทักษะ โดยมีรายละเอียดดังนี้

3.1) ทักษะด้านเทคนิคของนักพัฒนาซอฟต์แวร์

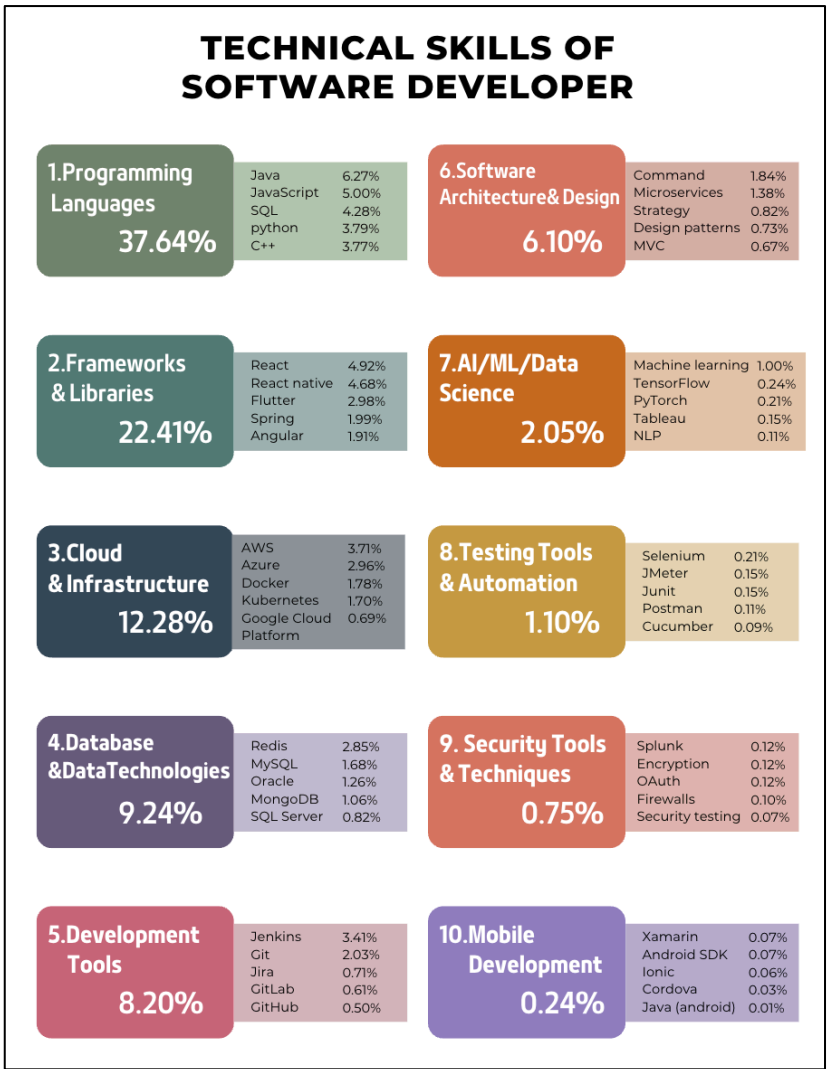
กลุ่มทักษะเชิงเทคนิคที่ปรากฏในประกาศรับสมัครงาน แสดงให้เห็นว่าตำแหน่งงานด้านซอฟต์แวร์ (เช่น Programmer หรือ Developer) ยังคงให้ความสำคัญอย่างมากกับทักษะที่เกี่ยวข้องกับการพัฒนาโค้ดโดยตรง ได้แก่ Programming Languages (37.64%) เป็นกลุ่มที่ได้รับความสำคัญสูงสุด รองลงมา Frameworks & Libraries (22.41%) เป็นกลุ่มที่มีบทบาทสำคัญรองลงมา ช่วยเพิ่มความเร็วและมาตรฐานในการพัฒนา ถัดมา Cloud & Infrastructure (12.28%) และ Database & Data Technologies (9.24%) แสดงถึงความต้องการนักพัฒนาที่สามารถ deploy ระบบบน cloud และจัดการฐานข้อมูลได้อย่างยืดหยุ่น Development Tools (8.20%) มีความสำคัญในด้าน workflow และ automation และSoftware Architecture & Design (6.10%) แสดงถึงความต้องการนักพัฒนาที่เข้าใจหลักการออกแบบระบบ เพื่อสนับสนุนการขยายตัวและการบำรุงรักษาระบบในระยะยาว ส่วน Other (4.13%) คือทักษะเฉพาะกลุ่ม เช่น Mobile Development, AI/ML, หรือ Security ที่เกี่ยวข้องกับบริบทงานเฉพาะทาง ดังภาพที่ 4-7



ภาพที่ 4-7 สัดส่วนทักษะด้านเทคนิคของนักพัฒนาซอฟต์แวร์

เมื่อพิจารณาทักษะย่อยในแต่ละกลุ่มทักษะที่ต้องมี 95% แรก โดยเลือกนำเสนอเฉพาะทักษะย่อยที่พบมากที่สุด 5 อันดับแรก ดังตารางที่ 4-18 พบว่า Programming Languages (37.64%) พบว่า นักพัฒนา

ซอฟต์แวร์ส่วนใหญ่จำเป็นต้องมีทักษะด้านการเขียนโปรแกรมได้แก่ Java, JavaScript, SQL, Python และ C++ ตามลำดับ ต่อมา Frameworks & Libraries (22.41%) พบว่า นักพัฒนาซอฟต์แวร์ส่วนใหญ่ควรมีทักษะด้านนี้ตามลำดับ ได้แก่ React, React native, Flutter และ Spring ตามลำดับส่วน Cloud & Infrastructure (12.28%) พบว่า ทักษะที่นักพัฒนาซอฟต์แวร์ควรมีได้แก่ AWS, Azure, Docker, Kubernetes, Google Cloud Platform ตามลำดับ Database & Data Technologies (9.24%) พบว่า ทักษะที่นักพัฒนาซอฟต์แวร์ควรมีได้แก่ Redis, MySQL, Oracle, MongoDB และ SQL Server ตามลำดับ และ Development Tools (8.20%) พบว่า ทักษะที่นักพัฒนาซอฟต์แวร์ควรมี ได้แก่ Jenkins, Git, Jira, GitLab และ GitHub ตามลำดับ แสดงดังภาพที่ 4-8

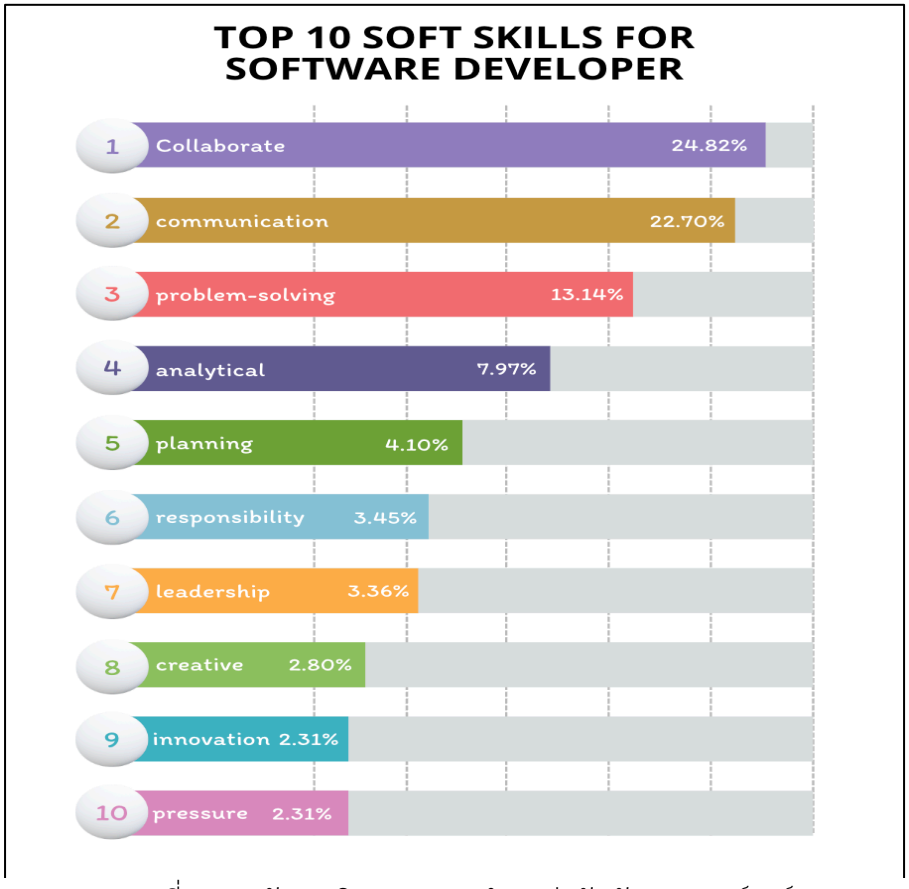


ภาพที่ 4-8 ทักษะเชิงเทคนิคของตำแหน่งนักพัฒนาซอฟต์แวร์

3.2) ทักษะด้านบุคคลของนักพัฒนาซอฟต์แวร์

จำนวนและร้อยละของทักษะเชิงบุคคลที่ปรากฏในประกาศรับสมัครงานของตำแหน่งนักพัฒนาซอฟต์แวร์ จำนวนทั้งสิ้น 34,076 รายการ พบว่า Collaborate (24.82%) เป็นทักษะที่ได้รับการกล่าวถึงมาก

ที่สุด สะท้อนถึงธรรมชาติของการพัฒนาซอฟต์แวร์ที่มักทำงานเป็นทีม การเขียนโค้ดร่วมกัน การ review code และการแก้ปัญหาร่วมกันในทีมแบบ Agile หรือ DevOps รองลงมา Communication (22.70%) เป็นทักษะรองลงมา ซึ่งเน้นความสามารถในการสื่อสารกับทีม พูดคุยกับผู้ใช้ (user) และรายงานผลการทำงานอย่างชัดเจน การสื่อสารที่ดีช่วยลดข้อผิดพลาดในกระบวนการพัฒนา ต่อมา Problem-solving (13.14%) และ Analytical (7.97%) แสดงให้เห็นถึงความสามารถในการคิดเชิงตรรกะและแก้ปัญหาทางเทคนิคที่เกิดขึ้นระหว่างการพัฒนา และ Planning (4.10%), Responsibility (3.45%), และ Leadership (3.36%) สะท้อนให้เห็นว่าบางตำแหน่งของนักพัฒนา (โดยเฉพาะ senior หรือ team lead) จำเป็นต้องมีทักษะด้านการวางแผน ควบคุมงาน และนำทีม ส่วนทักษะด้าน Creative (2.80%), Innovation (2.31%), และ Pressure (2.31%) แสดงให้เห็นว่าองค์กรยังให้ความสำคัญกับความสามารถในการสร้างสรรค์งานใหม่ การปรับตัว และการทำงานภายใต้แรงกดดัน โดยเฉพาะในโครงการที่เปลี่ยนแปลงเร็ว ดังภาพที่ 4-9



ภาพที่ 4-9 ทักษะเชิงบุคคลของตำแหน่งนักพัฒนาซอฟต์แวร์

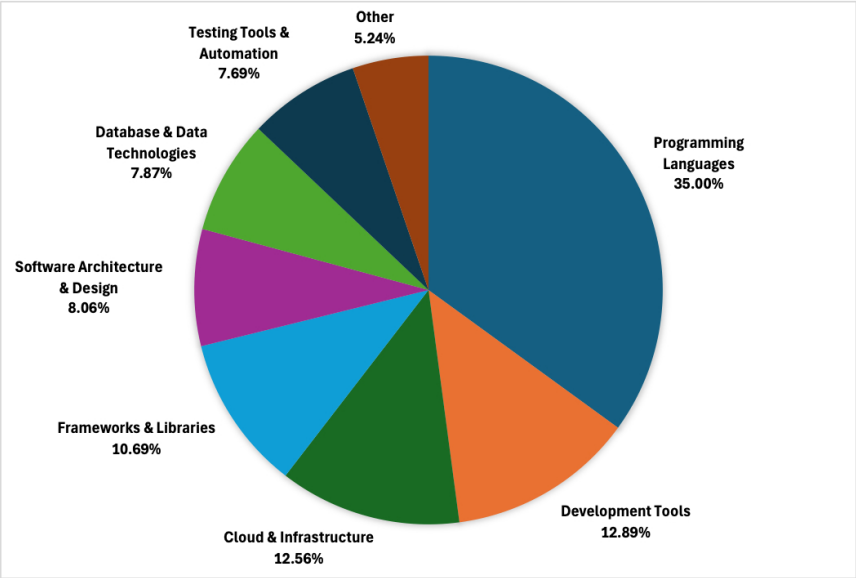
4) นักทดสอบซอฟต์แวร์ (Software Tester)

จากการรวบรวมประกาศรับสมัครงานทั้งหมดในตำแหน่งวิศวกรความต้องการ 70,893 รายการพบว่าผลการวิเคราะห์ทักษะเชิงเทคนิคที่ปรากฏในประกาศรับสมัครงานสำหรับตำแหน่งวิศวกรความต้องการ โดย

พิจารณาจากกลุ่มทักษะเชิงเทคนิค 10 กลุ่ม มีทักษะด้านเทคนิค จำนวน 65,535 ทักษะ และทักษะด้านบุคคล จำนวน 52,606 ทักษะ โดยมีรายละเอียดดังนี้

4.1) ทักษะด้านเทคนิคของนักทดสอบซอฟต์แวร์

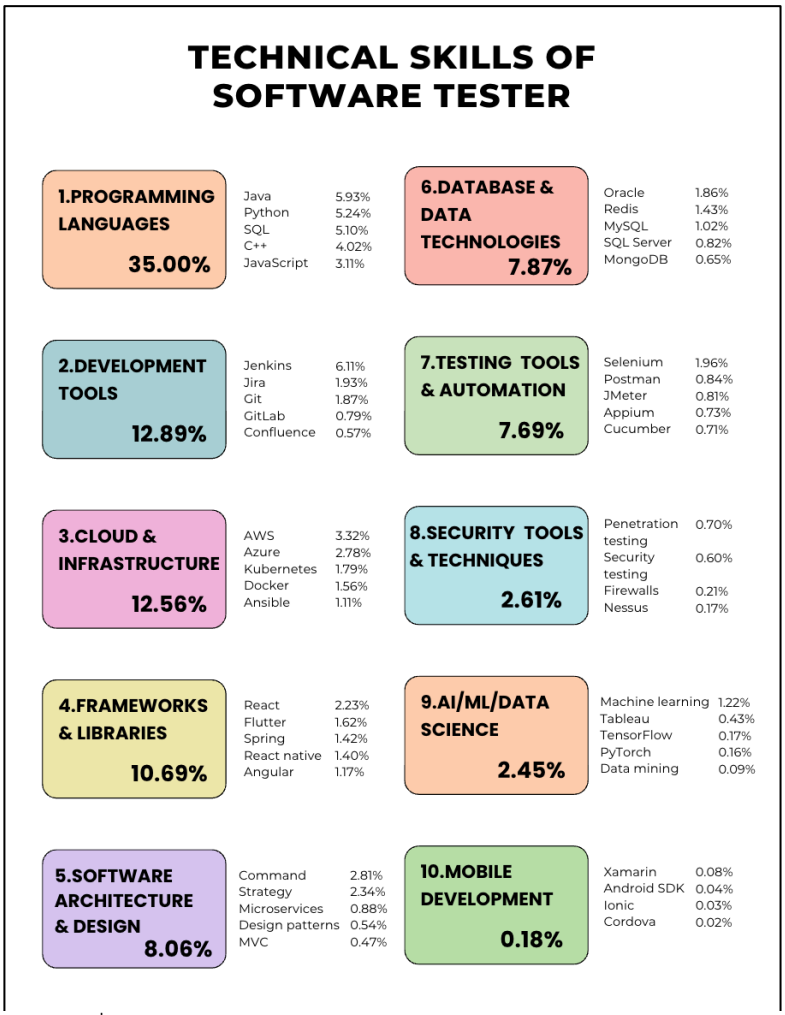
ผลการวิเคราะห์ พบว่า กลุ่มทักษะที่ปรากฏสูงสุดประมาณ 95% แรกอยู่ในกลุ่ม ได้แก่ Programming Languages (35.00%) รองลงมาคือกลุ่ม Development Tools (12.89%), Cloud & Infrastructure (12.56%) Frameworks & Libraries (10.69%), Software Architecture & Design (8.06%) และ ทักษะด้าน Database & Data Technologies (7.87%) ตามลำดับ ดังภาพที่ 4-10



ภาพที่ 4-10 สัดส่วนทักษะด้านเทคนิคของนักทดสอบซอฟต์แวร์

เมื่อพิจารณาทักษะย่อยในแต่ละกลุ่มทักษะที่ต้องมี 95% แรก โดยเลือกนำเสนอเฉพาะทักษะย่อยที่พบมากที่สุด 5 อันดับแรก ดังตารางที่ 4-20 พบว่า ทักษะที่ถูกระบุมากที่สุดคือ Programming Languages ซึ่งมีสัดส่วนสูงถึง 35.00% ของทั้งหมดแสดงให้เห็นว่านักทดสอบสมัยใหม่ต้องมีความรู้ด้านการเขียนโปรแกรม โดยเฉพาะสำหรับการสร้าง test scripts และการทำ automation testing โดยภาษา Java, Python, SQL และ C++ เป็นภาษาหลักที่พบมากที่สุด รองลงมา Development Tools (12.89%) เช่น Jenkins, Jira, Git เป็นเครื่องมือพื้นฐานในการบริหารจัดการ pipeline การทดสอบ การจัดการ issue และ version control ซึ่งสอดคล้องกับการทำงานแบบ DevOps/CI-CD ต่อมา Cloud & Infrastructure (12.56%) สะท้อนว่านักทดสอบซอฟต์แวร์จำนวนมากต้องทดสอบระบบที่รันอยู่บน cloud environment เช่น AWS, Azure หรือ Docker จึงต้องมีความเข้าใจด้านการติดตั้ง การจัดการ container และ cloud service ต่าง ๆ Frameworks & Libraries (10.69%) ทักษะด้าน React, Flutter, Spring และ Angular ถูกระบุในตำแหน่งนักทดสอบด้วย อาจสะท้อนถึงหน้าที่ในการทดสอบ frontend หรือระบบแบบ full-stack ที่ต้องเข้าใจโครงสร้างของเฟรมเวิร์กที่ใช้งานจริง

ส่วน Software Architecture & Design (8.06%) เป็นสิ่งสำคัญที่ช่วยให้นักทดสอบเข้าใจโครงสร้างของระบบ เพื่อสามารถออกแบบ test cases ได้ครอบคลุม เช่น understanding of microservices, MVC, หรือ design patterns และ Database & Data Technologies (7.87%) เช่น SQL, Oracle, MongoDB ช่วยให้นักทดสอบสามารถตรวจสอบข้อมูลใน backend ตรวจสอบเงื่อนไข และเขียน test cases ที่อิงกับฐานข้อมูลได้แม่นยำ และสุดท้าย Testing Tools & Automation (7.69%) ทักษะเฉพาะด้านที่สำคัญ เช่น Selenium, Postman, JMeter และ Appium เป็นเครื่องมือที่ใช้ในการทดสอบระบบอัตโนมัติ (automation testing), การทดสอบ API และการทดสอบประสิทธิภาพ ดังภาพที่ 4-11



ภาพที่ 4-11 ทักษะเชิงเทคนิคของตำแหน่งนักทดสอบซอฟต์แวร์

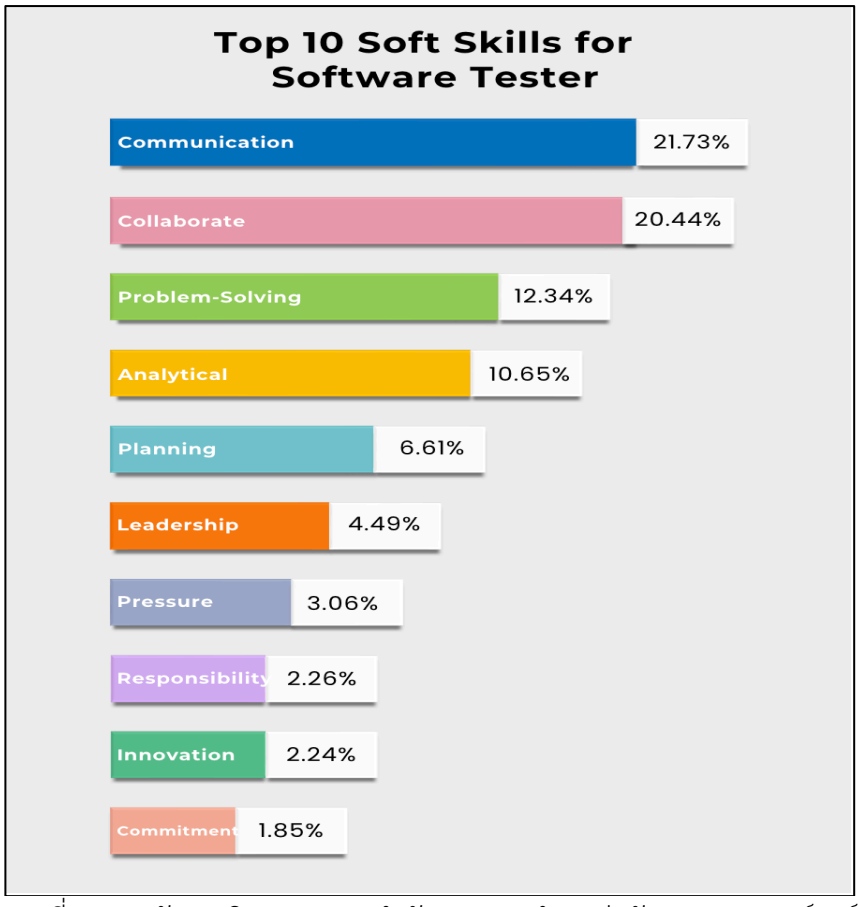
4.2) ทักษะด้านบุคคลของนักทดสอบซอฟต์แวร์

จากการวิเคราะห์ข้อมูลในตารางที่ 4-21 พบว่าตำแหน่งนักทดสอบซอฟต์แวร์ให้ความสำคัญกับทักษะเชิงบุคคลที่เกี่ยวข้องกับการสื่อสาร การทำงานร่วมกัน และการคิดวิเคราะห์ โดยมีรายละเอียดดังนี้ ทักษะที่พบมากที่สุด คือ Communication (21.73%) รองลง Collaborate (20.44%)

การทำงานร่วมกันในทีมทดสอบหรือกับทีมพัฒนา ต่อมา Problem-Solving (12.34%) การหาสาเหตุของข้อผิดพลาดและออกแบบกรณีทดสอบที่เหมาะสม ส่วนทักษะ Analytical (10.65%) นักทดสอบ

จำเป็นต้องวิเคราะห์สถานการณ์ต่าง ๆ อย่างแม่นยำ Planning (6.61%) ทักษะการวางแผนเกี่ยวข้องกับการจัดตารางการทดสอบ การเตรียมแผน test cases และการกำหนด test coverage ให้ครบถ้วน

ส่วนทักษะรองอื่นๆ ได้แก่ Leadership (4.49%) และ Responsibility (3.06%) สะท้อนถึงบทบาทของ tester ระดับ senior ที่ต้องดูแลกระบวนการหรือเป็นผู้ตรวจสอบคุณภาพให้กับทีม นอกจากนี้ ยังมีทักษะด้าน Pressure, Creative, และ Innovation รวมกันประมาณ 6.35% แสดงให้เห็นว่า tester ต้องสามารถทำงานภายใต้แรงกดดัน และมีความคิดสร้างสรรค์ในการออกแบบกรณีทดสอบให้ครอบคลุมทั้งเชิงบวกและเชิงลบ ดังภาพที่ 4-12



ภาพที่ 4-12 ทักษะเชิงบุคคล 10 ลำดับแรกของตำแหน่งนักทดสอบซอฟต์แวร์

5) วิศวกรซ่อมบำรุงซอฟต์แวร์ (Software Maintenance Engineer)

จากการรวบรวมประกาศรับสมัครงานทั้งหมดในตำแหน่งวิศวกรซ่อมบำรุง 31,825 รายการพบว่า ผลการวิเคราะห์ทักษะเชิงเทคนิคที่ปรากฏในประกาศรับสมัครงานสำหรับตำแหน่งวิศวกรความต้องการ โดยพิจารณาจากกลุ่มทักษะเชิงเทคนิค 10 กลุ่ม มีทักษะด้านเทคนิค จำนวน 22,370 ทักษะ และทักษะด้านบุคคลจำนวน 25,556 ทักษะ โดยมีรายละเอียดดังนี้

5.1) ทักษะด้านเทคนิคของวิศวกรซ่อมบำรุงซอฟต์แวร์

ผลการวิเคราะห์ พบว่า กลุ่มทักษะที่ปรากฏสูงสุดประมาณ 95% แรกอยู่ในกลุ่ม ได้แก่ จากแผนภาพวงกลมที่แสดงสัดส่วนทักษะเชิงเทคนิคของตำแหน่ง Software Maintenance Engineer พบว่า กลุ่มทักษะที่ปรากฏในประกาศรับสมัครงานมีความหลากหลายและครอบคลุมหลายด้าน โดยสามารถสรุปผลได้ดังนี้:

กลุ่มทักษะหลักที่ปรากฏเด่นชัด

Programming Languages (28.31%) ซึ่งเป็นกลุ่มที่มีสัดส่วนมากที่สุด สะท้อนให้เห็นถึงความสำคัญของทักษะการเขียนโปรแกรมในงานซ่อมบำรุงซอฟต์แวร์ เช่น การแก้บั๊ก การปรับปรุงโค้ด และการรองรับความเข้ากันได้ของระบบในเวอร์ชันใหม่

Cloud & Infrastructure (26.73%) เป็นกลุ่มทักษะที่รองลงมา บ่งชี้ว่างานซ่อมบำรุงในปัจจุบันมักเกี่ยวข้องกับระบบที่รันบน cloud environment เช่น AWS, Azure หรือ Docker ดังนั้นผู้ปฏิบัติงานจำเป็นต้องเข้าใจการ deploy และ monitor ระบบบนคลาวด์อย่างมีประสิทธิภาพ

กลุ่มทักษะรองที่สำคัญ

Database & Data Technologies (11.17%) ทักษะนี้มีความสำคัญอย่างมากในการซ่อมแซมและปรับปรุงระบบที่เกี่ยวข้องกับฐานข้อมูล เช่น การแก้ไขปัญหาข้อมูลไม่ถูกต้อง การปรับ performance query หรือการดูแล schema เดิมให้เหมาะสมกับระบบที่เปลี่ยนไป

Software Architecture & Design (10.88%) แสดงให้เห็นว่าผู้ปฏิบัติงานในตำแหน่งนี้ต้องมีความเข้าใจโครงสร้างระบบซอฟต์แวร์ในภาพรวม เช่น Microservices, Design Patterns, MVC เพื่อสามารถปรับปรุงได้อย่างเป็นระบบและไม่กระทบต่อโมดูลอื่น

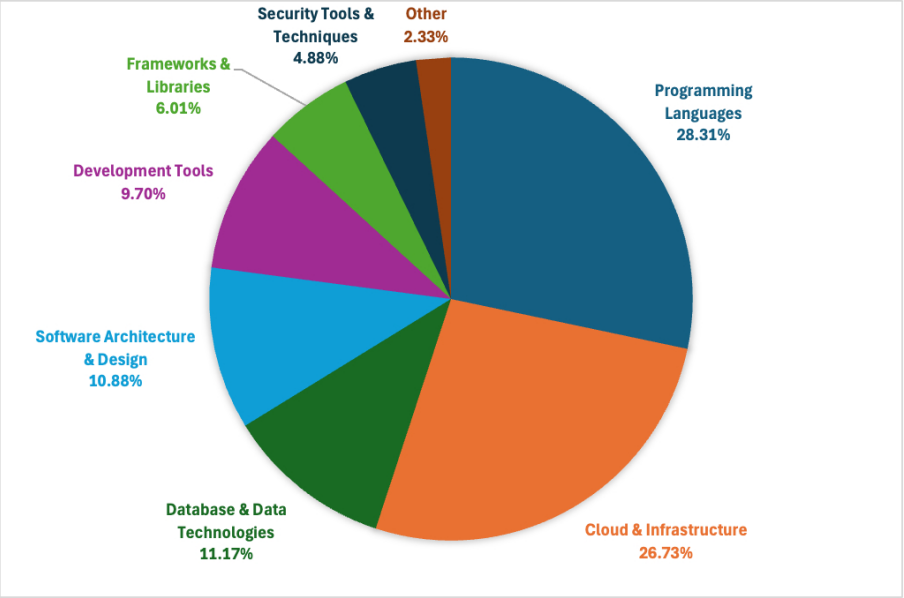
Development Tools (9.70%) เช่น Git, Jenkins, Jira ซึ่งเป็นเครื่องมือที่ใช้ควบคุมเวอร์ชัน, จัดการงาน และทำ CI/CD ช่วยให้งานซ่อมบำรุงสามารถทำงานร่วมกับทีมพัฒนาได้อย่างราบรื่น

กลุ่มทักษะสนับสนุนเพิ่มเติม

Frameworks & Libraries (6.01%) ชี้ให้เห็นถึงความจำเป็นในการเข้าใจเทคโนโลยีที่ระบบถูกพัฒนาขึ้น เช่น Spring, React, Flutter ซึ่งช่วยให้สามารถ debug หรือ update ได้ตรงจุด

Security Tools & Techniques (4.88%) แม้จะมีสัดส่วนน้อยกว่า แต่สะท้อนว่าตำแหน่งนี้ยังคงต้องดูแลด้านความปลอดภัยของระบบ เช่น การอัปเดต patch ความปลอดภัย หรือการตรวจสอบช่องโหว่

และ Other (2.33%) รวมถึงทักษะเฉพาะทางอื่น ๆ ที่มีความจำเป็นในบางบริบทของงาน เช่น การดูแล legacy systems, การเขียน test scripts หรือการใช้เครื่องมือ automation testing ขั้นพื้นฐาน ดังภาพที่ 4 -13

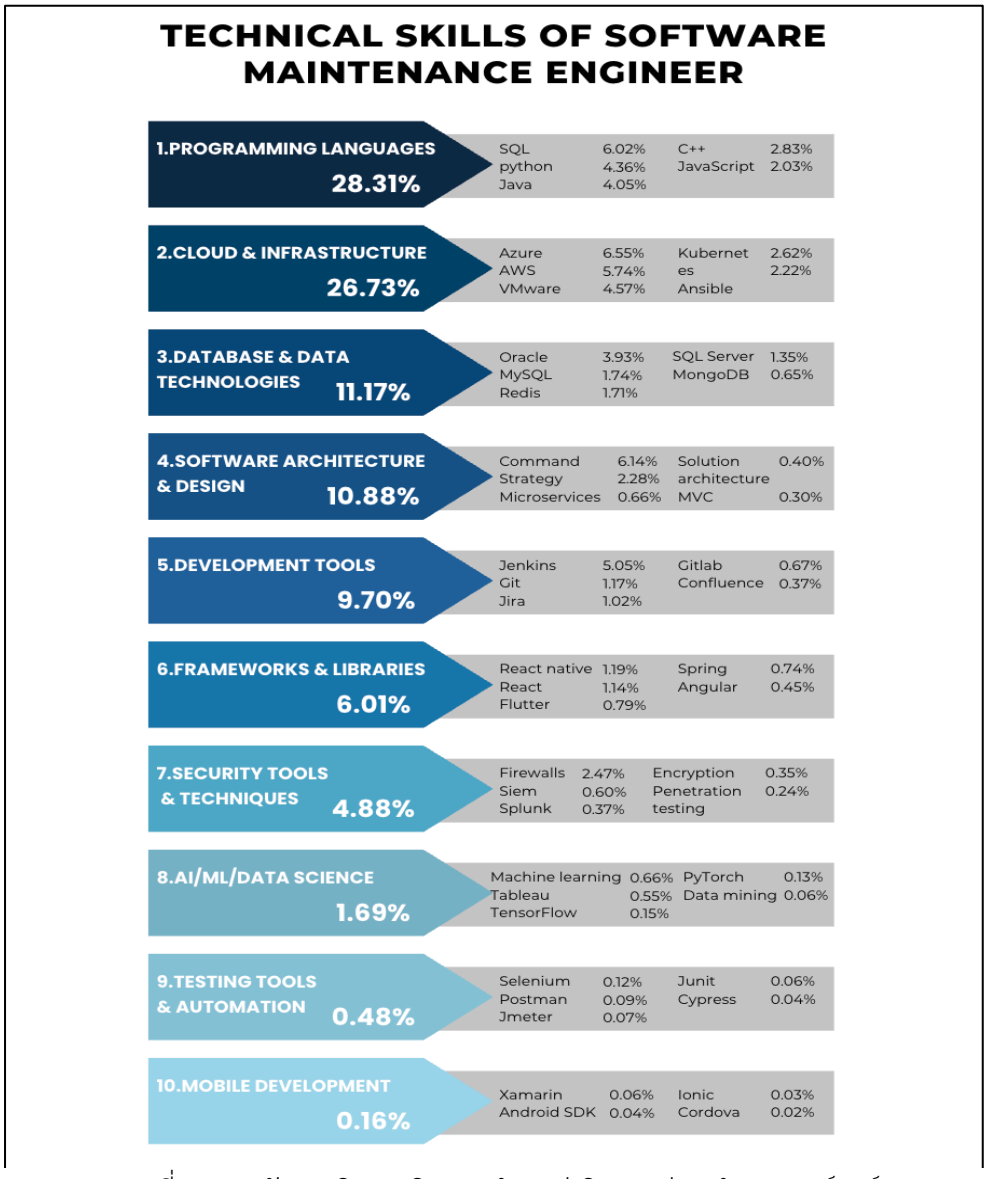


ภาพที่ 4-13 สัดส่วนทักษะด้านเทคนิคของวิศวกรซ่อมบำรุงซอฟต์แวร์

เมื่อพิจารณาทักษะย่อยในแต่ละกลุ่มทักษะที่ต้องมี 95% แรก โดยเลือกนำเสนอเฉพาะทักษะย่อยที่พบมากที่สุด 5 อันดับแรก ดังตารางที่ 4-22 แสดงให้เห็นว่าตำแหน่งวิศวกรซ่อมบำรุงซอฟต์แวร์มีความต้องการทักษะเชิงเทคนิคหลากหลาย โดยเฉพาะทักษะด้านการเขียนโปรแกรม การจัดการโครงสร้างระบบ และฐานข้อมูล ทั้งนี้สามารถแจกแจงในแต่ละกลุ่มได้ดังนี้:

กลุ่มทักษะที่มีความต้องการสูงที่สุด ได้แก่ Programming Languages (28.31%) ทักษะด้านภาษาโปรแกรมเป็นพื้นฐานของการทำงานซ่อมบำรุงซอฟต์แวร์ ไม่ว่าจะเป็นการแก้บั๊ก ปรับปรุงฟังก์ชัน หรือต่อยอดระบบเดิม โดยภาษา SQL, Python และ Java เป็นภาษาที่พบมากที่สุดในประกาศงาน สะท้อนถึงความจำเป็นในงานดูแลระบบฐานข้อมูลและระบบ back-end รองลงมา Cloud & Infrastructure (26.73%) ซึ่งการที่ทักษะกลุ่มนี้มีสัดส่วนสูงเป็นอันดับสอง สะท้อนว่าองค์กรต้องการบุคลากรที่เข้าใจระบบโครงสร้างพื้นฐานสมัยใหม่ ไม่ว่าจะเป็น Azure, AWS หรือ VMware เนื่องจากระบบซอฟต์แวร์ในปัจจุบันส่วนใหญ่อยู่บน cloud และต้องการการดูแลระบบที่ซับซ้อน

กลุ่มทักษะที่สนับสนุนการซ่อมบำรุง ได้แก่ Database & Data Technologies (11.17%) การดูแลระบบฐานข้อมูลมีบทบาทสำคัญในงานซ่อมบำรุง โดยเฉพาะการปรับปรุง query หรือการตรวจสอบข้อมูลผิดพลาด ทักษะที่พบได้บ่อย ได้แก่ Oracle, MySQL และ Redis ซึ่งใช้ในระบบเดิมและระบบใหม่ควบคู่กัน รองลงมา Software Architecture & Design (10.88%) ทักษะกลุ่มนี้ช่วยให้วิศวกรสามารถเข้าใจภาพรวมของโครงสร้างซอฟต์แวร์เพื่อปรับเปลี่ยนหรือแก้ไขระบบได้อย่างไม่กระทบต่อส่วนอื่น เช่น การใช้ design patterns แบบ Command, Strategy หรือ MVC ซึ่งยังคงเป็นโครงสร้างหลักในระบบ enterprise และ Development Tools (9.70%) ซึ่งเครื่องมือสนับสนุนการพัฒนาระบบ เช่น Jenkins, Git, และ Jira มีความสำคัญต่อการบริหารเวอร์ชัน การ deploy อัตโนมัติ และการจัดการงานร่วมกับทีม ดังภาพที่ 4-14



ภาพที่ 4-14 ทักษะเชิงเทคนิคของตำแหน่งวิศวกรซ่อมบำรุงซอฟต์แวร์

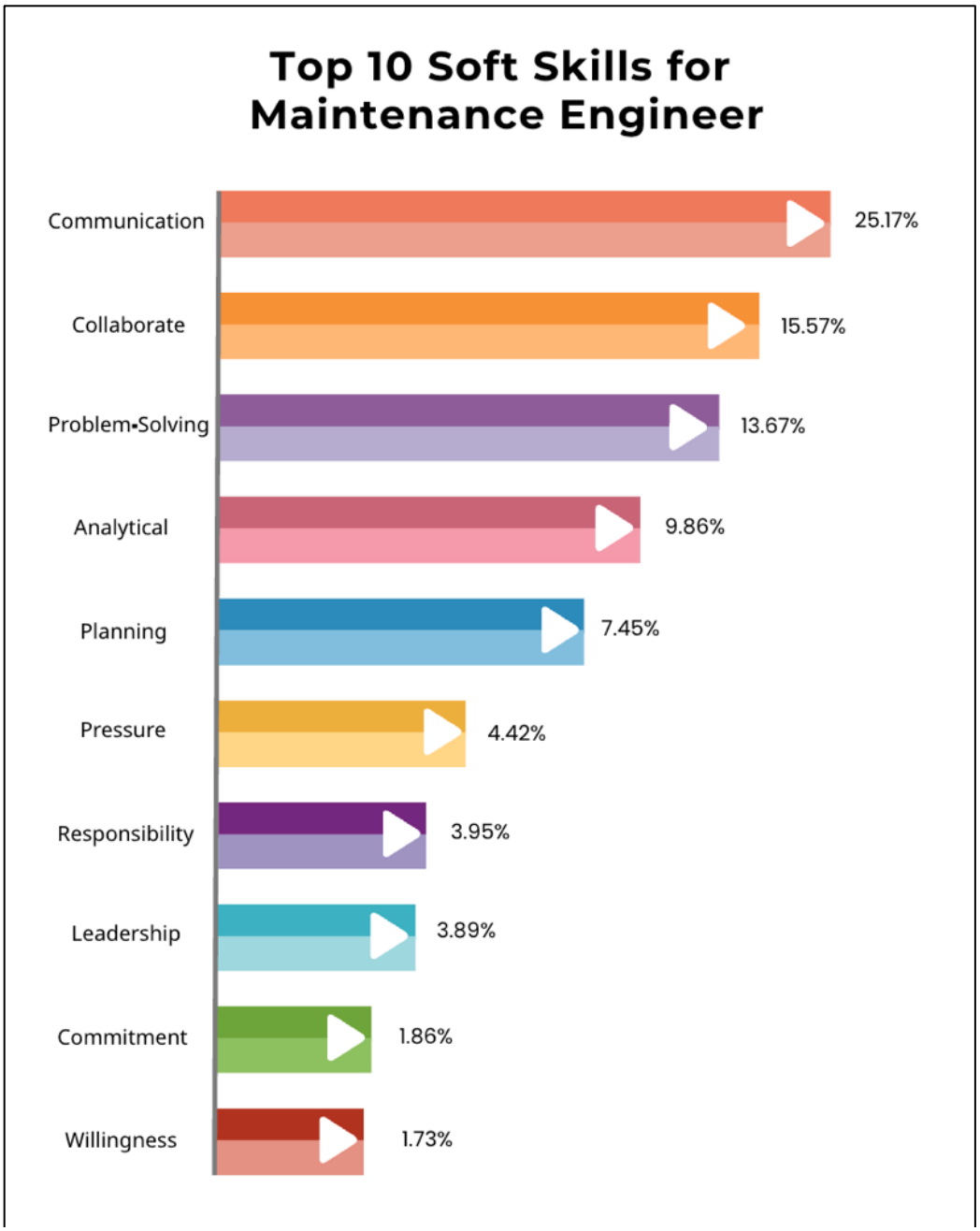
5.2) ทักษะด้านบุคคลของตำแหน่งวิศวกรซ่อมบำรุงซอฟต์แวร์

จากการวิเคราะห์ข้อมูลทักษะเชิงบุคคลที่ปรากฏในประกาศรับสมัครงานตำแหน่งวิศวกรซ่อมบำรุงซอฟต์แวร์ พบว่าทักษะเหล่านี้มีความสำคัญควบคู่กับทักษะทางเทคนิค โดยมีรายละเอียดดังนี้

ทักษะที่พบมากที่สุด ได้แก่ Communication (25.17%) ซึ่งเป็นทักษะการสื่อสารเป็นองค์ประกอบที่สำคัญที่สุดในตำแหน่งนี้ รองลงมา Collaborate (15.57%) ความสามารถในการทำงานร่วมกับผู้อื่นการร่วมมือระหว่างนักพัฒนาและผู้ทดสอบ และ Problem-Solving (13.67%) ซึ่งเป็นทักษะการแก้ปัญหาเป็นสิ่งที่ขาดไม่ได้ในตำแหน่งนี้

ทักษะสนับสนุนด้านกระบวนการคิด ได้แก่ Analytical (9.86%) ซึ่งเป็นทักษะการคิดเชิงวิเคราะห์ช่วยให้สามารถแยกแยะสาเหตุของปัญหาได้อย่างแม่นยำ รองลงมา Planning (7.45%) ทักษะการวางแผนช่วยให้การจัดลำดับความสำคัญของงานซ่อมบำรุงมีความเป็นระบบ ส่วนทักษะด้านจิตวิทยาและคุณลักษณะส่วน

บุคคล Pressure (4.42%) สะท้อนให้เห็นว่าผู้ปฏิบัติงานต้องสามารถทำงานภายใต้แรงกดดัน Responsibility (3.95%) ความรับผิดชอบเป็นสิ่งที่จำเป็นเพราะมีผลกระทบโดยตรงต่อการทำงานของระบบจริง ส่วน Leadership, Commitment, Willingness (รวม ~7.48%) แม้จะมีสัดส่วนน้อย แต่สะท้อนให้เห็นถึงความคาดหวังในความเป็นผู้นำ ความตั้งใจ และความเต็มใจในการพัฒนาระบบอย่างต่อเนื่อง ดังภาพที่ 4-15



ภาพที่ 4-15 ทักษะเชิงบุคคลของตำแหน่งวิศวกรซ่อมบำรุงซอฟต์แวร์

6) ผู้จัดการโครงการซอฟต์แวร์ (Software Project Manager)

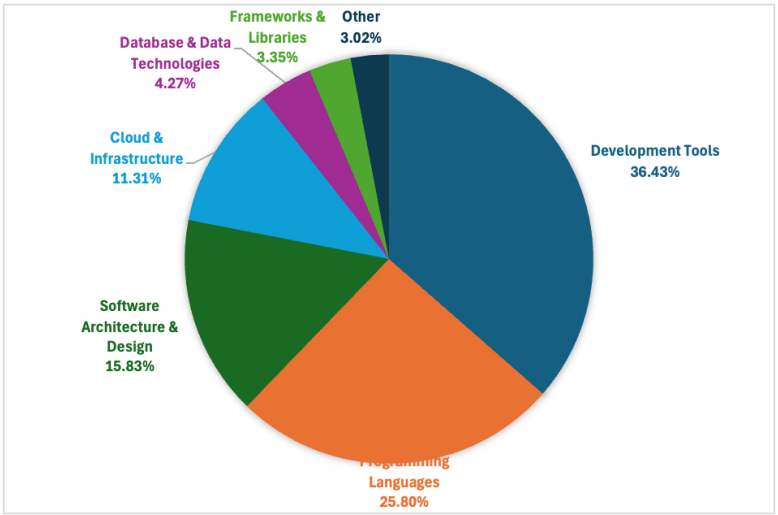
จากการรวบรวมประกาศรับสมัครงานทั้งหมดในตำแหน่งวิศวกรซ่อมบำรุง 1,868 รายการพบว่า ผลการวิเคราะห์ทักษะเชิงเทคนิคที่ปรากฏในประกาศรับสมัครงานสำหรับตำแหน่งวิศวกรความต้องการ โดยพิจารณาจากกลุ่มทักษะเชิงเทคนิค 10 กลุ่ม มีทักษะด้านเทคนิค จำนวน 2,388 ทักษะ และทักษะด้านบุคคล จำนวน 11,768 ทักษะ โดยมีรายละเอียดดังนี้

6.1) ทักษะด้านเทคนิคของผู้จัดการโครงการซอฟต์แวร์

จากการวิเคราะห์ประกาศรับสมัครงานในตำแหน่งผู้จัดการโครงการซอฟต์แวร์ พบว่ามีการระบุทักษะเชิงเทคนิคในกลุ่มต่าง ๆ โดยเรียงตามสัดส่วนความถี่ที่พบมากที่สุด ดังนี้

Development Tools (36.43%) เป็นกลุ่มทักษะที่พบมากที่สุดในตำแหน่งนี้ สะท้อนให้เห็นถึงความสำคัญของการใช้เครื่องมือในการวางแผน ควบคุมเวอร์ชัน และติดตามความก้าวหน้าของโครงการ รองลงมา Programming Languages (25.80%) แม้ผู้จัดการโครงการจะไม่ได้เขียนโค้ดโดยตรง แต่การเข้าใจภาษาที่ทีมใช้งาน ต่อมา Software Architecture & Design (15.83%) เน้นความเข้าใจในโครงสร้างระบบและแนวทางการออกแบบซอฟต์แวร์ เพื่อให้สามารถบริหารจัดการขอบเขตของงาน ฟังก์ชัน และการบูรณาการระบบได้อย่างถูกต้อง Cloud & Infrastructure (11.31%) สะท้อนถึงความจำเป็นที่ผู้จัดการโครงการต้องเข้าใจเทคโนโลยีพื้นฐาน เพื่อวางแผนการใช้งบประมาณ และจัดสรรทรัพยากรด้านโครงสร้างพื้นฐานของโครงการได้

ส่วน Database & Data Technologies (4.27%) ความเข้าใจพื้นฐานด้านฐานข้อมูล เช่น SQL, Oracle, MongoDB เป็นประโยชน์ต่อการติดตามการจัดการข้อมูลหรือวางแผนจัดเก็บข้อมูลในระบบต่าง ๆ Frameworks & Libraries (3.35%) ทักษะนี้อาจไม่ได้ใช้งานโดยตรง แต่หากผู้จัดการเข้าใจโครงสร้างของ Frameworks ที่ทีมใช้ (เช่น React, Spring) ก็จะช่วยในการกำหนดขอบเขตและเวลาได้แม่นยำยิ่งขึ้น และ Other (3.02%) กลุ่มทักษะอื่น ๆ ที่ปรากฏในจำนวนที่น้อยกว่ากลุ่มหลัก ดังภาพที่ 4-16



ภาพที่ 4-16 สัดส่วนทักษะด้านเทคนิคของดการโครงการซอฟต์แวร์

เมื่อพิจารณาทักษะย่อยในแต่ละกลุ่มทักษะที่ต้องมี 95% แรก โดยเลือกนำเสนอเฉพาะทักษะย่อยที่พบมากที่สุด 5 อันดับแรก ดังภาพที่ 4-17 พบว่า

กลุ่ม Development Tools (ร้อยละ 36.43%) เป็นกลุ่มทักษะที่ปรากฏมากที่สุดในตำแหน่งนี้ สะท้อนว่าผู้จัดการโครงการจำเป็นต้องมีความสามารถในการใช้เครื่องมือสำหรับติดตามงานและการประสานงานทีมพัฒนา โดยเครื่องมือที่พบมากที่สุดคือ Jira (20.35%) และ Confluence (7.20%) ซึ่งใช้ในการบริหารโครงการและเอกสารความรู้ รวมถึง Trello (5.53%) สำหรับงาน Kanban และ Jenkins/Slack เพื่อสนับสนุนการพัฒนาแบบอัตโนมัติและการสื่อสารภายในทีม

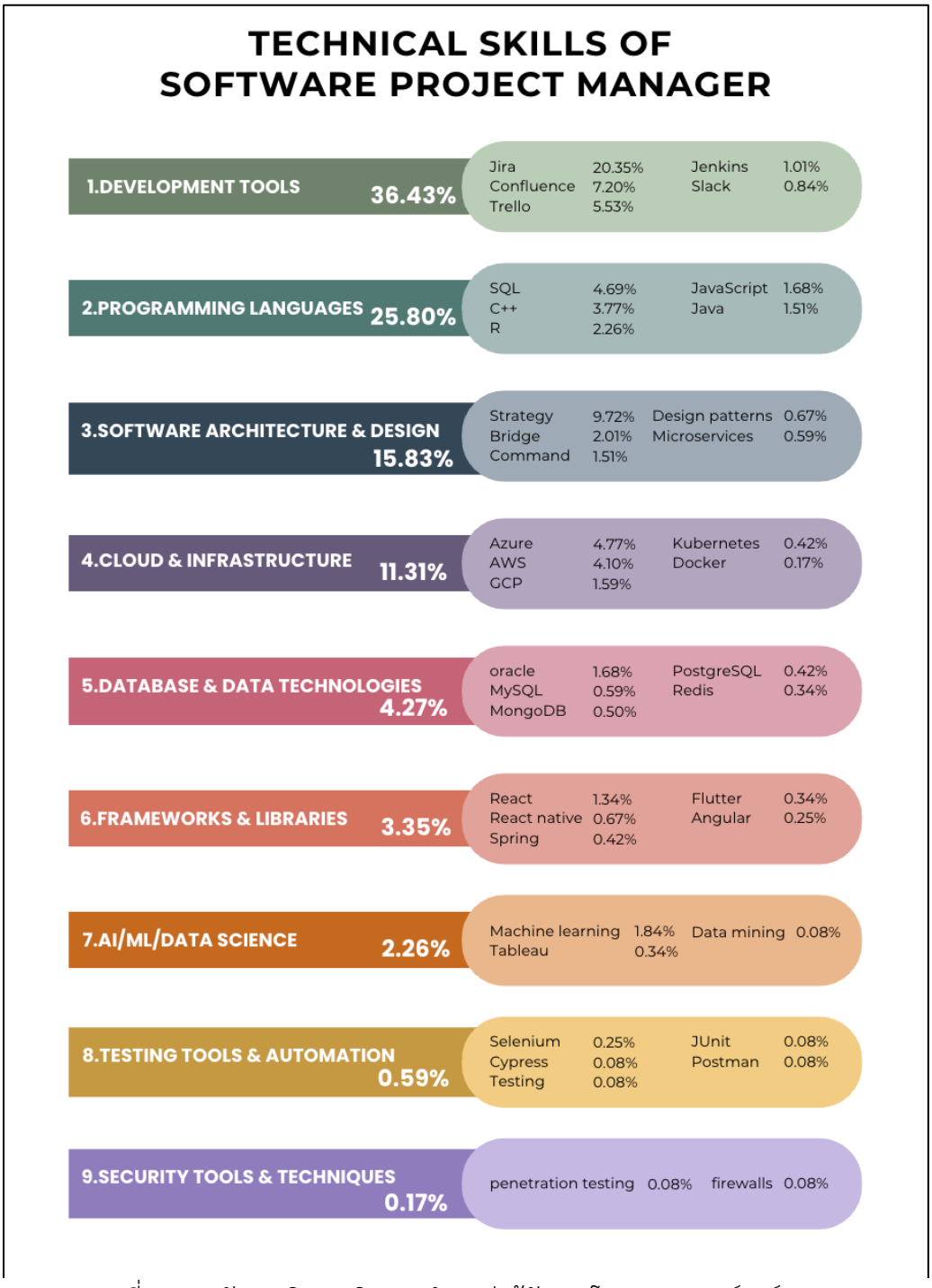
กลุ่ม Programming Languages (ร้อยละ 25.80%) แม้จะไม่ใช่งานเขียนโค้ดโดยตรง แต่ผู้จัดการโครงการที่มีพื้นฐานด้านการเขียนโปรแกรมจะช่วยให้สามารถเข้าใจข้อจำกัดและระยะเวลาในการพัฒนาได้อย่างแม่นยำ โดยเฉพาะภาษาที่นิยม ได้แก่ SQL, C++, R, JavaScript และ Java

กลุ่ม Software Architecture & Design (ร้อยละ 15.83%) แสดงถึงความต้องการให้ผู้จัดการโครงการมีความเข้าใจในการออกแบบระบบ โดยเฉพาะ Strategy (9.72%), Bridge, และ Command Pattern ซึ่งเป็นแนวคิดสำคัญในการจัดการโครงสร้างงานพัฒนาให้มีความยืดหยุ่น

กลุ่ม Cloud & Infrastructure (ร้อยละ 11.31%) สะท้อนถึงความจำเป็นในการบริหารจัดการโครงการที่เกี่ยวข้องกับระบบ Cloud หรือการ Deploy โดยเฉพาะผู้จัดการโครงการที่เกี่ยวข้องกับ DevOps หรือ SaaS เครื่องมือสำคัญที่พบ ได้แก่ Azure, AWS, GCP, Kubernetes และ Docker

กลุ่ม Database & Data Technologies (ร้อยละ 4.27%) เป็นทักษะฐานข้อมูลยังคงมีบทบาทในงานของผู้จัดการโครงการในกรณีที่เกี่ยวข้องกับการออกแบบระบบข้อมูลหรือการวิเคราะห์ความต้องการของผู้ใช้งาน

กลุ่ม Frameworks & Libraries (ร้อยละ 3.35%) ถึงแม้จะไม่ใช่กลุ่มทักษะหลักของผู้จัดการโครงการโดยตรง แต่การมีความรู้พื้นฐานเกี่ยวกับ Framework ที่ทีมใช้งาน เช่น React, Spring หรือ Angular จะช่วยให้สามารถสนับสนุนการตัดสินใจและสื่อสารกับทีมพัฒนาได้อย่างมีประสิทธิภาพ

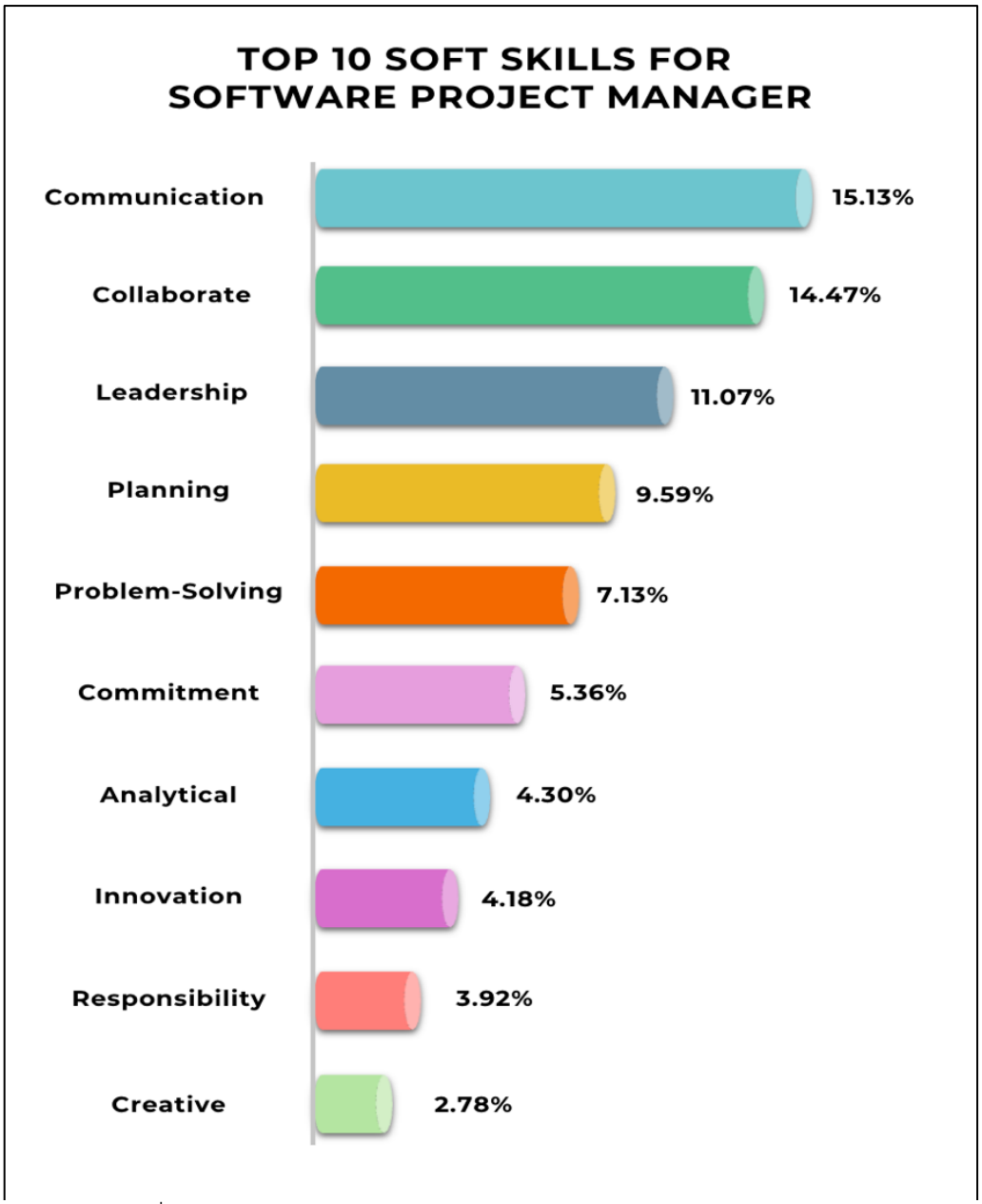


ภาพที่ 4-17 ทักษะเชิงเทคนิคของตำแหน่งผู้จัดการโครงการซอฟต์แวร์

6.2) ทักษะด้านบุคคลของตำแหน่งผู้จัดการโครงการ

จากการรวบรวมข้อมูลประกาศรับสมัครงาน พบว่า ทักษะเชิงบุคคล (Soft Skills) มีบทบาทสำคัญอย่างมากในตำแหน่งผู้จัดการโครงการซอฟต์แวร์ โดยเฉพาะทักษะที่เกี่ยวข้องกับการประสานงาน การบริหารทีม และการวางแผนโครงการ ดังนี้ ทักษะด้านการสื่อสารถูกระบุมากที่สุด Communication (ร้อยละ 15.13%) รองลงมา Collaborate (ร้อยละ 14.47%) ซึ่งเป็นความสามารถในการทำงานร่วมกับผู้อื่น ต่อมา Leadership

(ร้อยละ 11.07%) ผู้จัดการโครงการจำเป็นต้องมีภาวะผู้นำ Planning (ร้อยละ 9.59%) เป็นทักษะสำคัญในการกำหนด Timeline, Task Breakdown, Resource Allocation และการจัดการความเสี่ยงในโครงการ ส่วน Problem-Solving (ร้อยละ 7.13%) สะท้อนความสามารถของผู้จัดการโครงการในการวิเคราะห์สถานการณ์ และตัดสินใจแก้ปัญหา Commitment (ร้อยละ 5.36%) องค์กรให้ความสำคัญกับความมุ่งมั่นและความรับผิดชอบของผู้จัดการโครงการในการขับเคลื่อนโครงการจนสำเร็จ ส่วนอื่นๆ ได้แก่ Analytical, Innovation, Responsibility และ Creative แสดงให้เห็นถึงการคาดหวังให้ผู้จัดการโครงการมีทักษะเชิงวิเคราะห์ คิดเชิงนวัตกรรม มีความรับผิดชอบ และความคิดสร้างสรรค์ ซึ่งช่วยส่งเสริมการจัดการโครงการในยุคที่เทคโนโลยีเปลี่ยนแปลงรวดเร็ว ดังภาพที่ 4-18



ภาพที่ 4-18 ทักษะเชิงบุคคล 10 อันดับแรกของตำแหน่งผู้จัดการโครงการ

7) นักวิเคราะห์ระบบ (System Analyst)

จากการรวบรวมประกาศรับสมัครงานทั้งหมดในตำแหน่งวิศวกรซ่อมบำรุง 90,529 รายการพบว่า ผลการวิเคราะห์ทักษะเชิงเทคนิคที่ปรากฏในประกาศรับสมัครงานสำหรับตำแหน่งวิศวกรความต้องการ โดยพิจารณาจากกลุ่มทักษะเชิงเทคนิค 10 กลุ่ม มีทักษะด้านเทคนิค จำนวน 76,561 ทักษะ และทักษะด้านบุคคล จำนวน 69,392 ทักษะ โดยมีรายละเอียดดังนี้

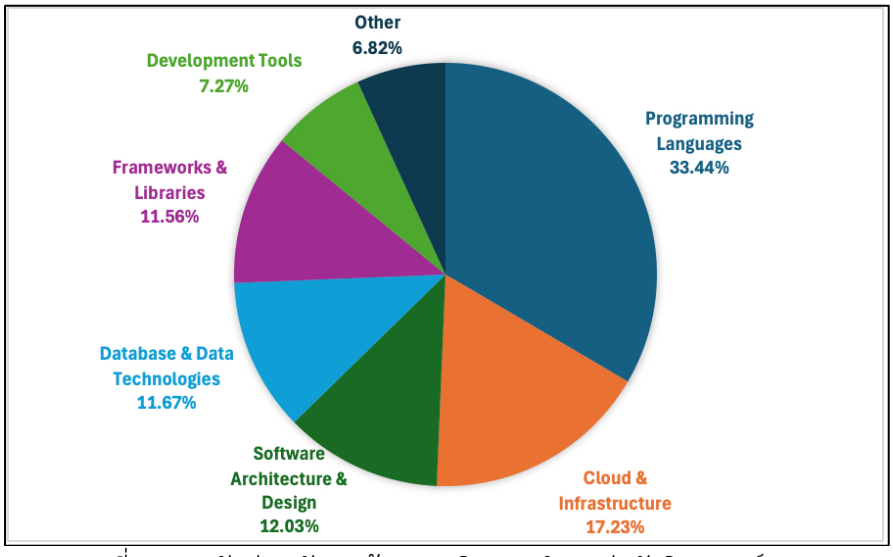
7.1) ทักษะด้านเทคนิคของตำแหน่งนักวิเคราะห์ระบบ

จากการรวบรวมข้อมูลประกาศรับสมัครงาน พบว่า ทักษะเชิงเทคนิค (Technical Skills) ที่ปรากฏในตำแหน่งนักวิเคราะห์ระบบ (System Analyst) มีความหลากหลาย โดยมีการกระจุกตัวของความต้องการในกลุ่ม Programming และ Cloud Infrastructure เป็นหลัก ดังนี้

Programming Languages (ร้อยละ 33.44%) เป็นทักษะที่ปรากฏสูงสุด โดยองค์กรต้องการ System Analyst ที่สามารถเข้าใจและเขียนโค้ดได้ เพื่อให้สามารถวิเคราะห์หรือเข้าใจระบบได้อย่างลึกซึ้ง ทักษะนี้ช่วยในการสื่อสารกับฝ่ายพัฒนาและตรวจสอบความถูกต้องของระบบได้อย่างแม่นยำ

รองลงมา Cloud & Infrastructure (ร้อยละ 17.23%) สะท้อนให้เห็นว่าบทบาทของ System Analyst มีความเกี่ยวข้องกับโครงสร้างพื้นฐานของระบบและสถาปัตยกรรมคลาวด์มากขึ้น เช่น การวิเคราะห์ความต้องการด้านโฮสติ้ง ความสามารถในการขยายตัว และการเชื่อมโยงระบบผ่าน Cloud Service ต่าง ๆ Software Architecture & Design (ร้อยละ 12.03%) ซึ่งเป็นทักษะด้านการออกแบบสถาปัตยกรรมซอฟต์แวร์เป็นสิ่งที่องค์กรให้ความสำคัญ โดยเฉพาะในระบบที่มีความซับซ้อน จำเป็นต้องมีผู้วิเคราะห์ระบบที่สามารถออกแบบโครงสร้างที่เหมาะสมและปรับปรุงประสิทธิภาพของระบบได้

ส่วน Database & Data Technologies (ร้อยละ 11.67%) System Analyst ต้องสามารถเข้าใจโครงสร้างฐานข้อมูล และสามารถออกแบบหรือดึงข้อมูลเพื่อวิเคราะห์ความต้องการหรือวางโครงสร้างระบบได้ Frameworks & Libraries (ร้อยละ 11.56%) แสดงถึงการคาดหวังให้ Analyst เข้าใจเทคโนโลยีที่ทีมพัฒนานิยมใช้งาน เพื่อให้สามารถวิเคราะห์และวางแผนการพัฒนาร่วมกับทีมได้อย่างมีประสิทธิภาพ และ Development Tools (ร้อยละ 7.27%) สะท้อนถึงการใช้งานเครื่องมือที่สนับสนุนการพัฒนาซอฟต์แวร์ เช่น Git, Jira, Jenkins ซึ่งช่วยให้ Analyst ติดตามความคืบหน้าของงานและวิเคราะห์ปัญหาในกระบวนการพัฒนาได้ ส่วนทักษะอื่นๆ (ร้อยละ 6.82%) อาจรวมถึงเครื่องมือเฉพาะทางหรือภาษาพัฒนาอื่น ๆ ซึ่งพบได้น้อยแต่มีความเฉพาะเจาะจงตามลักษณะงานของแต่ละองค์กร ดังภาพที่ 4-19



ภาพที่ 4-19 สัดส่วนทักษะด้านเทคนิคของตำแหน่งนักวิเคราะห์ระบบ

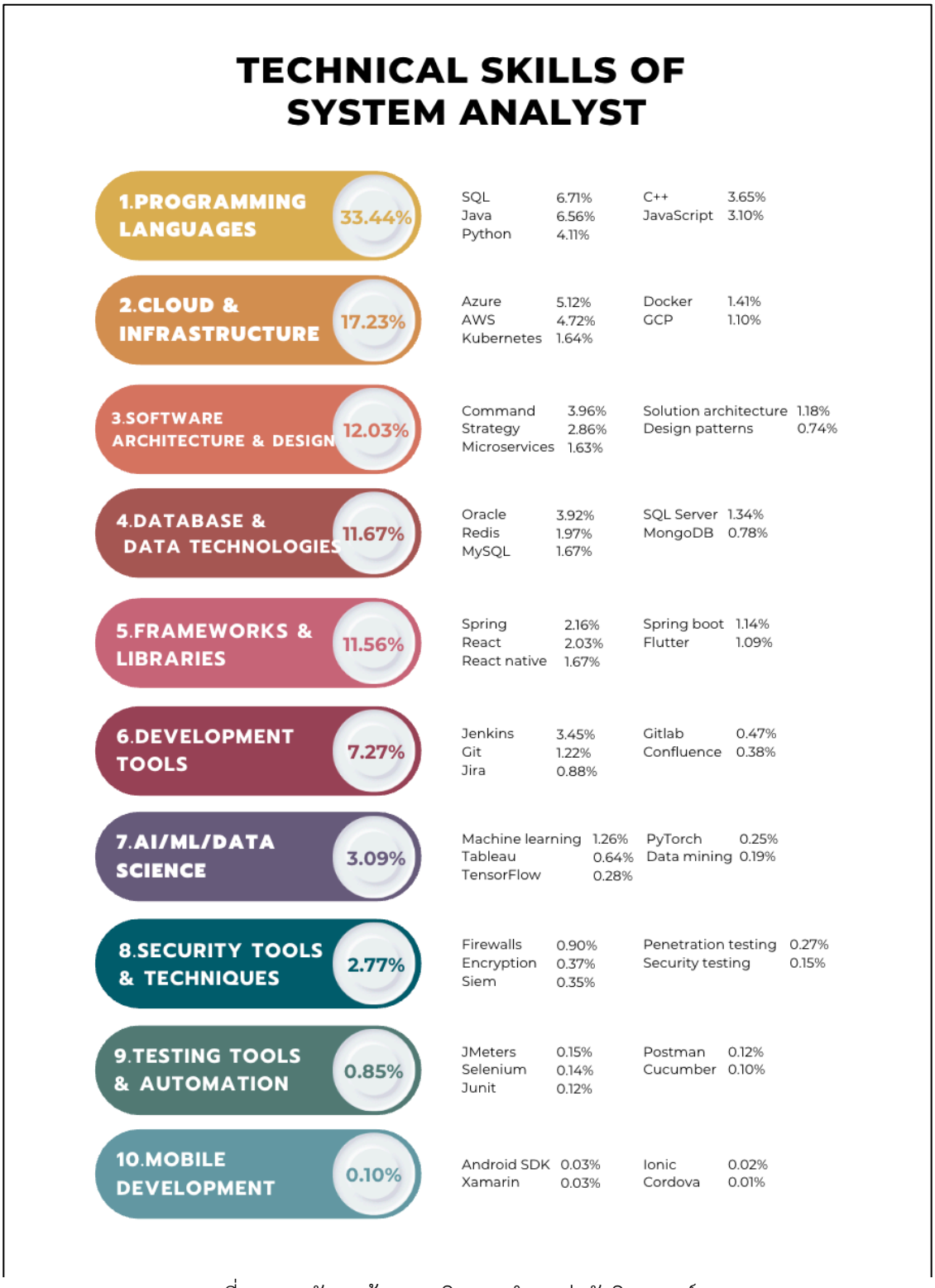
เมื่อพิจารณาทักษะย่อยในแต่ละกลุ่มทักษะที่ต้องมี 95% แรก โดยเลือกนำเสนอเฉพาะทักษะย่อยที่พบมากที่สุด 5 อันดับแรก ดังตารางที่ 4-26 พบว่า

กลุ่มทักษะที่ปรากฏสูงสุด คือ Programming Languages (33.44%) โดยเฉพาะภาษาที่นิยมในงานวิเคราะห์ระบบ เช่น SQL (6.71%), Java (6.56%), Python และ C++ สะท้อนให้เห็นว่าผู้วิเคราะห์ระบบจำเป็นต้องเข้าใจโค้ดเพื่อสื่อสารกับทีมพัฒนา ตีความความต้องการ และออกแบบระบบได้อย่างแม่นยำ รองลงมา Cloud & Infrastructure (17.23%) ระบบสมัยใหม่มักพัฒนาบนโครงสร้าง Cloud เช่น Azure และ AWS ซึ่ง Analyst จำเป็นต้องเข้าใจหลักการทำงานและบริการพื้นฐานเพื่อรองรับการออกแบบระบบให้มีความยืดหยุ่นและขยายตัวได้ในอนาคต Software Architecture & Design (12.03%) ทักษะด้านการออกแบบสถาปัตยกรรมซอฟต์แวร์ เช่น Command, Strategy, Microservices เป็นสิ่งสำคัญต่อการวางโครงสร้างระบบที่สอดคล้องกับความต้องการเชิงธุรกิจ

Database & Data Technologies (11.67%) ซึ่งเป็นความสามารถในการทำงานกับฐานข้อมูล เช่น Oracle, MySQL, Redis ถือเป็นทักษะหลักของ System Analyst ในการวิเคราะห์ความสัมพันธ์ของข้อมูลและออกแบบฐานข้อมูลให้เหมาะสมกับระบบ

Frameworks & Libraries (11.56%) ทักษะเกี่ยวกับเครื่องมือและไลบรารี เช่น Spring, React, Flutter มีบทบาทสำคัญในการเข้าใจองค์ประกอบของซอฟต์แวร์ เพื่อให้วิเคราะห์หรือวางแผนการพัฒนาร่วมกับทีมได้แม่นยำยิ่งขึ้น

Development Tools (7.27%) แม้จะไม่ได้เป็นทักษะแกนหลัก แต่เครื่องมือสนับสนุนการพัฒนา เช่น Jenkins, Git, Jira ก็มีความจำเป็นอย่างยิ่งสำหรับผู้วิเคราะห์ระบบในการติดตามเวอร์ชันโค้ดหรือการประสานงานในโครงการ ดังภาพที่ 4-19



ภาพที่ 4-20 ทักษะด้านเทคนิคของตำแหน่งนักวิเคราะห์ระบบ

7.2) ทักษะด้านบุคคลของตำแหน่งนักวิเคราะห์ระบบ

การศึกษาประกาศรับสมัครงานแสดงให้เห็นว่า ตำแหน่งนักวิเคราะห์ระบบมีความต้องการทักษะเชิงบุคคลที่หลากหลาย โดยเน้นทักษะที่ช่วยสนับสนุนการทำงานร่วมกับทีมและการคิดวิเคราะห์เชิงลึกเพื่อให้สามารถแปลงความต้องการทางธุรกิจให้เป็นระบบงานได้อย่างมีประสิทธิภาพ โดยมีรายละเอียดดังนี้

1. Communication (ร้อยละ 22.69) ทักษะการสื่อสารเป็นอันดับหนึ่งของความต้องการในตำแหน่งนี้ เพราะนักวิเคราะห์ระบบต้องถ่ายทอดความต้องการของผู้ใช้งานไปยังทีมพัฒนาอย่างถูกต้อง และสื่อสารข้อมูลทางเทคนิคให้อีกฝ่ายเข้าใจได้ง่าย

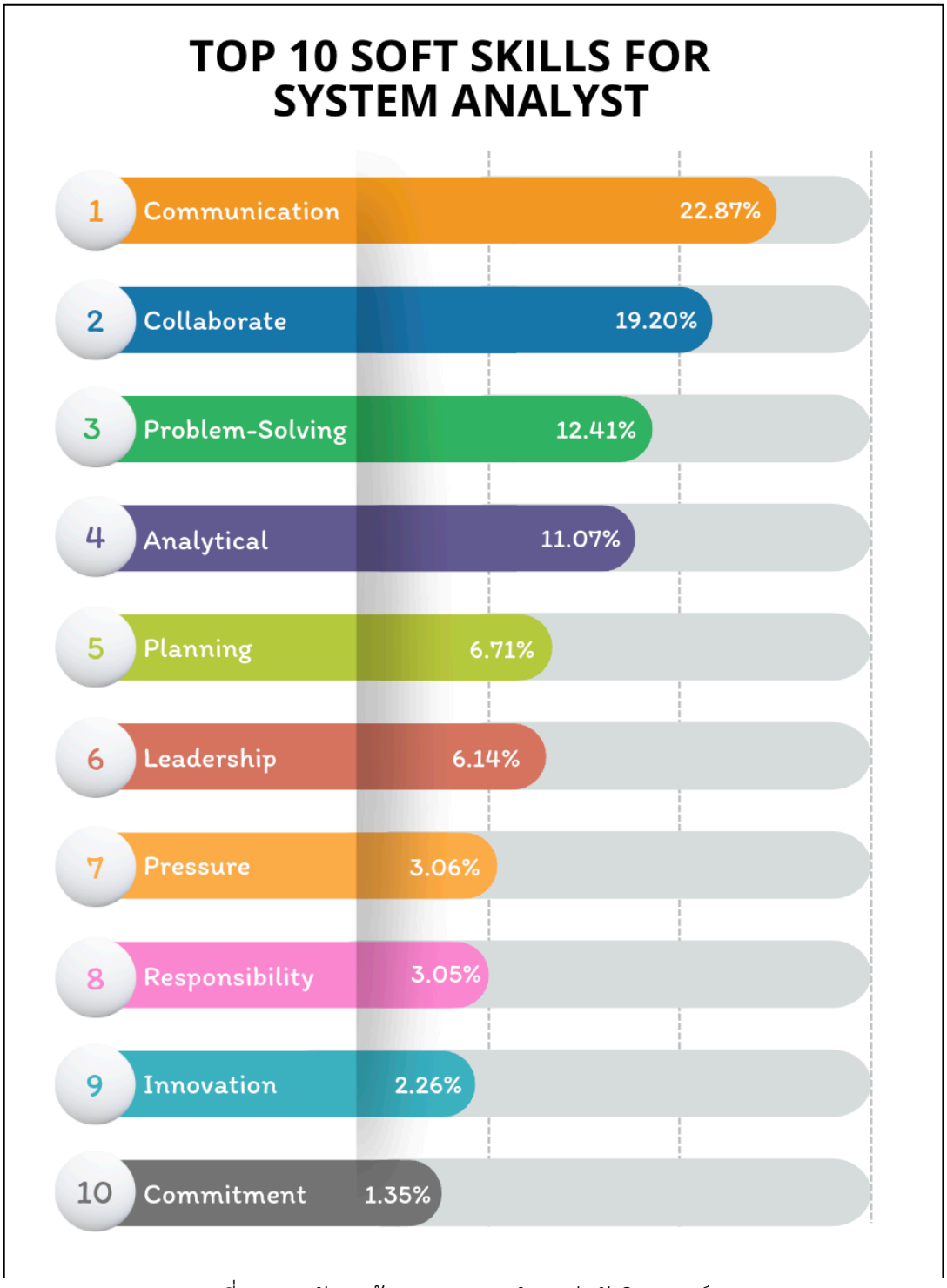
2. Collaborate (ร้อยละ 19.20) ความสามารถในการทำงานร่วมกับผู้อื่น (ทั้งทีมพัฒนา ผู้ใช้งาน และผู้บริหาร) มีความสำคัญอย่างมาก นักวิเคราะห์ระบบมักเป็นศูนย์กลางของกระบวนการพัฒนา จึงต้องสามารถประสานงานและทำงานร่วมกับคนหลายบทบาทได้อย่างราบรื่น

3. Problem-Solving (ร้อยละ 12.41) นักวิเคราะห์ระบบจำเป็นต้องมีทักษะในการวิเคราะห์ปัญหาทางธุรกิจ และแปลงเป็นโซลูชันที่สามารถพัฒนาเป็นระบบได้อย่างเหมาะสม ทักษะนี้จึงมีความสำคัญในเชิงความคิดเชิงวิเคราะห์และตรรกะ

4. Analytical (ร้อยละ 11.07) เน้นความสามารถในการแยกแยะข้อมูล ความต้องการ และกระบวนการเพื่อกำหนดโครงสร้างระบบได้อย่างมีประสิทธิภาพ เป็นทักษะพื้นฐานของงานวิเคราะห์ระบบ

5. Planning (ร้อยละ 6.71) การวางแผนงานให้ครอบคลุมระยะเวลาพัฒนาและความต้องการของโครงการ เป็นอีกหนึ่งความสามารถที่ช่วยให้นักวิเคราะห์ระบบมีบทบาทสนับสนุนการทำงานร่วมกับ Project Manager ได้ดีขึ้น

โดยรายละเอียดในแต่ละทักษะด้านบุคคล ดังภาพที่ 4-20



ภาพที่ 4-21 ทักษะด้านบุคคลของตำแหน่งนักวิเคราะห์ระบบ

4.3 ตัวแบบการทำนายตำแหน่งงานตามกลุ่มทักษะตามอาชีพด้วยเทคนิคการจำแนกประเภท (Classification techniques)

การดำเนินการเพื่อสร้างตัวแบบ (model) สำหรับทำนายตำแหน่งงานจากกลุ่มทักษะที่ปรากฏในประกาศรับสมัครงาน โดยใช้เทคนิคการจำแนกประเภท (Classification) ได้แก่ Decision Tree, Random Forest, Naïve Bayes, XGBoost และ ซึ่งงานวิจัยนี้ใช้กระบวนการ Train-Test Split (80:20) และประเมินผลด้วยค่า Accuracy, Precision, Recall และ F1-score ซึ่งได้ผลลัพธ์ ดังตารางที่ 4-14

ตารางที่ 4-14 ผลลัพธ์การเปรียบเทียบประสิทธิภาพของโมเดล

Model	Accuracy	Precision	Recall	F1-score
Random Forest	0.620	0.618	0.620	0.619
Naïve Bayes	0.598	0.590	0.596	0.593
XGBoost	0.603	0.600	0.601	0.600
LSTM (Deep learning)	0.574	0.570	0.574	0.572

ตารางที่ 4-14 แสดงผลการเปรียบเทียบประสิทธิภาพของโมเดลที่ใช้ในการจำแนกข้อมูล โดยใช้ค่าชี้วัด 4 ตัว ได้แก่ Accuracy, Precision, Recall และ F1-score จากผลลัพธ์พบว่า Random Forest ให้ประสิทธิภาพสูงที่สุด โดยมีค่า Accuracy เท่ากับ 0.620 และ F1-score เท่ากับ 0.619 ซึ่งแสดงถึงความสมดุลระหว่าง Precision และ Recall โมเดล XGBoost มีค่าที่ใกล้เคียงกับ Random Forest เล็กน้อย รองลงมาเป็น Naïve Bayes ซึ่งให้ผลลัพธ์ในระดับปานกลาง ขณะที่ LSTM (Deep Learning) มีค่าต่ำที่สุดในทุกตัวชี้วัด สะท้อนว่าอาจยังไม่เหมาะสมกับชุดข้อมูลนี้ โดยสรุป โมเดลประเภท Machine Learning แบบดั้งเดิม เช่น Random Forest อาจมีความเหมาะสมมากกว่า Deep Learning สำหรับการวิเคราะห์ข้อมูลข้อความที่มีโครงสร้างในงานวิจัยนี้

บทที่ 5
สรุปและอธิปรายผล

5.1 สรุปผลการวิจัย

งานวิจัยนี้ต้องการศึกษาข้อมูลทักษะที่ต้องการขององค์กรซอฟต์แวร์ที่ประกาศรับตำแหน่งงานด้านซอฟต์แวร์จากเว็บไซต์รับสมัครงานชั้นนำในประเทศไทยเพื่อนำเสนอชุดของทักษะตามความต้องการในแต่ละตำแหน่งงานและสร้างตัวแบบสำหรับประเมินตำแหน่งงานที่เหมาะกับสมรรถนะในตำแหน่งงานที่กำหนด โดยนำข้อมูลประกาศงานที่เป็นภาษาอังกฤษ จากเว็บไซต์รับสมัครงานในประเทศไทย ได้แก่ Jobthai.com และ Linkedln.com ซึ่งตำแหน่งงานที่ผู้วิจัยสนใจเป็นตำแหน่งงานทางด้านวิศวกรรมซอฟต์แวร์ได้แก่ วิศวกรความต้องการ นักออกแบบซอฟต์แวร์ นักพัฒนาซอฟต์แวร์ นักทดสอบซอฟต์แวร์ วิศวกรซ่อมบำรุง ผู้จัดการโครงการ และนักวิเคราะห์ระบบ ภายใต้การพิจารณาทักษะตามความต้องการสองกลุ่มคือ กลุ่มทักษะด้านเทคนิค และกลุ่มทักษะด้านบุคคล ซึ่งได้ผลลัพธ์พบว่า ในประกาศงานทั้งหมดหลังการทำความสะอาดข้อมูลแล้ว ตำแหน่งที่มีประกาศมากที่สุดคือ System Analyst จำนวน 90,529 รายการ รองลงมาคือ Developer และ Software Tester ขณะที่ตำแหน่งที่มีจำนวนน้อยที่สุดคือ Project Manager ในด้าน Hard skill ตำแหน่ง Developer มีจำนวนคำสำคัญสูงที่สุดถึง 100,175 คำ ส่วนด้าน Soft skill พบว่าตำแหน่ง System Analyst มีจำนวนคำสำคัญสูงที่สุดที่ 69,392 คำ รองลงมาคือ Software Tester และ Developer ซึ่งสะท้อนถึงความสำคัญของทักษะด้านการสื่อสาร การวิเคราะห์ และการทำงานร่วมกับผู้อื่น

เมื่อพิจารณาในแต่ละตำแหน่งงานพบว่า ตำแหน่งวิศวกรความต้องการ กลุ่มทักษะที่ปรากฏสูงสุด 90% แรกอยู่ในกลุ่ม ได้แก่ Programming Languages (37.26%) รองลงมาคือกลุ่ม Software Architecture & Design (14.07%) Database & Data Technologies (12.38%) ส่วนทักษะด้านบุคคล คือ Communication (23.33%) รองลงมา Collaborate (16.84%) และ Problem-Solving (12.80%)

ตำแหน่งนักออกแบบซอฟต์แวร์ พบว่า กลุ่มทักษะด้านเทคนิคมากที่สุด ได้แก่ Programming Languages (38.30%) รองลงมาคือกลุ่ม Frameworks & Libraries (18.21%) และ Database & Data Technologies (12.16%) ส่วนทักษะด้านบุคคล คือ Collaborate รองลงมา Communication และ ทักษะ Problem-Solving

ตำแหน่งนักพัฒนาซอฟต์แวร์ พบว่า กลุ่มทักษะเชิงเทคนิคที่ปรากฏมากที่สุด ได้แก่ Programming Languages (37.64%) รองลงมา Frameworks & Libraries (22.41%) และ Cloud & Infrastructure (12.28%) ส่วนทักษะด้านบุคคล พบว่า ทักษะที่พบมากที่สุด Collaborate (24.82%) รองลงมา Communication (22.70%) และ Problem-solving (13.14%)

ตำแหน่งนักทดสอบซอฟต์แวร์ พบว่าทักษะที่พบมากที่สุด คือ Programming Languages (35.00%) รองลงมาคือกลุ่ม Development Tools (12.89%) และ Cloud & Infrastructure (12.56%) ส่วนทักษะ

ด้านบุคคล พบว่าทักษะที่มากที่สุด Communication (21.73%) รองลง Collaborate (20.44%) และ Problem-Solving (12.34%)

ตำแหน่งวิศวกรซ่อมบำรุง พบว่า ทักษะที่พบมากที่สุด คือ Programming Languages (28.31%) รองลงมา Cloud & Infrastructure (26.73%) และ ทักษะในกลุ่ม Database & Data Technologies (11.17%) และทักษะด้านบุคคล พบว่า Communication (25.17%) รองลงมา Collaborate (15.57%) และ Problem-Solving (13.67%)

ตำแหน่งผู้จัดการโครงการ พบว่า Development Tools (36.43%) รองลงมา Programming Languages (25.80%) และ ทักษะ Software Architecture & Design (15.83%) ส่วนทักษะด้านบุคคล ได้แก่ Communication (ร้อยละ 15.13%) รองลงมา Collaborate (ร้อยละ 14.47%) และLeadership (ร้อยละ 11.07%)

ตำแหน่งนักวิเคราะห์ระบบ พบว่า ทักษะ Programming Languages (ร้อยละ 33.44%) เป็นทักษะที่ปรากฏสูงสุด รองลงมา Cloud & Infrastructure (ร้อยละ 17.23%) และ Software Architecture & Design (ร้อยละ 12.03%) ส่วนทักษะด้านบุคคล พบว่า Communication (ร้อยละ 22.69) รองลงมา Collaborate (ร้อยละ 19.20) และ Problem-Solving (ร้อยละ 12.41)

จะเห็นว่า Programming Languages เป็นทักษะทางเทคนิคที่ปรากฏมากที่สุดในเกือบทุกตำแหน่ง เช่น RE, SD, Dev, ST, ME, SA ส่วนทักษะด้านบุคคลที่พบเหมือนกันในเกือบทุกตำแหน่งคือ ทักษะ Communication ทักษะ Collaborate และ ทักษะ Problem-Solving ตำแหน่งงานส่วนใหญ่เน้นทักษะ Framework, Cloud, Development Tools, Software Architecture แตกต่างกันไปตามบทบาท แต่ยังอยู่ในกลุ่มทักษะหลักที่ใช้ร่วมกัน

ส่วน Project Manager (PM) จะเน้น Development Tools (36.43%) มากกว่าทุกตำแหน่ง และเป็นเพียงตำแหน่งเดียวที่ Leadership อยู่ในทักษะบุคคลใน 3 ลำดับแรก ส่วน Maintenance Engineer (ME) จะเน้น Cloud & Infrastructure (26.73%) รองจาก Programming Languages ตำแหน่ง Software Designer (SD) และ Developer (Dev) มีสัดส่วนของ Frameworks & Libraries สูงกว่าตำแหน่งอื่น และตำแหน่ง System Analyst (SA) มีความหลากหลายของทักษะสูง และปรากฏ Software Architecture & Design สูงกว่าเฉลี่ย

นอกจากนี้ผู้วิจัยศึกษาโมเดลสำหรับการทำนายตำแหน่งงานจากทักษะที่ได้เก็บรวบรวมมาเพื่อทำการเปรียบเทียบโมเดลการทำนายที่ดีที่สุดในกลุ่มของ โมเดล Machine learning ได้แก่ Random Forest Naïve Bayes XGBoost และโมเดล Deep learning ได้แก่ LSTM พบว่า Random Forest ให้ประสิทธิภาพสูงที่สุด และโมเดล XGBoost มีค่าที่ใกล้เคียงกับ Random Forest เล็กน้อย รองลงมาเป็น Naïve Bayes ซึ่งให้ผลลัพธ์ในระดับปานกลาง ขณะที่ LSTM (Deep Learning) มีค่าต่ำที่สุดในทุกตัวชี้วัด สะท้อนว่าอาจยังไม่เหมาะสมกับชุดข้อมูลนี้ โดยสรุป โมเดลประเภท Machine Learning แบบดั้งเดิม เช่น Random Forest อาจมีความเหมาะสมมากกว่า Deep Learning สำหรับการวิเคราะห์ข้อมูลข้อความที่มีโครงสร้างในงานวิจัยนี้

5.2 ข้อเสนอแนะงานวิจัย

ข้อมูลทักษะที่ต้องการขององค์กรซอฟต์แวร์ที่ประกาศรับตำแหน่งงานด้านซอฟต์แวร์จากเว็บไซต์รับสมัครงานชั้นนำในประเทศไทยเพื่อนำเสนอชุดของทักษะตามความต้องการในแต่ละตำแหน่งงานและสร้างตัวแบบสำหรับประเมินตำแหน่งงานที่เหมาะกับสมรรถนะในตำแหน่งงานที่กำหนด ผลการศึกษาดังกล่าวทำให้ผู้วิจัยได้ข้อเสนอเพื่อนำไปใช้ในอนาคต ได้แก่

1) ควรปรับหลักสูตรให้สอดคล้องกับความต้องการของตลาดแรงงาน โดยเฉพาะทักษะด้าน Programming Languages รวมถึงเครื่องมือ Frameworks และ DevOps ที่เป็นที่ต้องการสูง

2) หลักสูตรควรเสริมสร้างทักษะด้าน Soft Skills อย่างจริงจัง โดยเฉพาะ Communication, Collaborate และ Problem-Solving เนื่องจากพบว่าเป็นทักษะที่ถูกระบุในเกือบทุกตำแหน่งงาน และเป็นหัวใจสำคัญของการทำงานในทีม

2) ปรับปรุงการใช้เทคนิค NLP และ Machine Learning ในการวิเคราะห์ข้อมูลตำแหน่งงาน รวมถึงพัฒนาโมเดลที่แม่นยำขึ้น รวมถึงการวิเคราะห์ที่รองรับข้อมูลภาษาไทย

3) พัฒนาระบบวัดทักษะของบัณฑิตหลังจบการศึกษาเพื่อสะท้อนให้เห็นว่าบัณฑิตมีทักษะตามความต้องการของตลาดอุตสาหกรรมซอฟต์แวร์ในประเทศไทยตามข้อมูลจากการวิเคราะห์

5.3 ข้อจำกัดของงานวิจัย

1) แม้ว่าประกาศงานจะมาจากเว็บไซต์ในประเทศไทย แต่เนื่องจากข้อมูลทั้งหมดเป็นภาษาอังกฤษ ทำให้โมเดลที่พัฒนาขึ้นอาจไม่สามารถนำไปใช้ในบริบทของภาษาไทยได้โดยตรง ซึ่งมีโครงสร้างทางภาษาที่แตกต่างกันอย่างชัดเจน เช่น การทำให้เป็นตัวอักษรพิมพ์เล็ก การตัดโดยใช้ช่องว่าง และอักขระพิเศษเป็นต้น

2) การศึกษาวิจัยนี้ใช้เฉพาะตำแหน่งงาน 7 ตำแหน่ง เช่น Developer, Tester, Project Manager เป็นต้น ซึ่งอาจไม่เพียงพอในการสะท้อนภาพรวมของอุตสาหกรรมซอฟต์แวร์ทั้งหมดที่มีตำแหน่งงานที่มีตำแหน่งงานที่หลากหลายกว่านี้

3) การจำแนกทักษะในแต่ละตำแหน่งที่มีทักษะหลายอย่างซ้ำซ้อนกัน อาจจะทำให้การทำนายตำแหน่งงานมีความแม่นยำต่ำ

4) การใช้เทคนิค SMOTE อาจเป็นวิธีที่จะช่วยปรับสมดุลข้อมูลได้ดีในระดับหนึ่ง แต่การสังเคราะห์ข้อมูลใหม่จากคลาสจำนวนน้อยอาจก่อให้เกิด noise หรือ bias ในบางตำแหน่งที่มีลักษณะทักษะเฉพาะตัวสูง

5) โมเดลวิเคราะห์จากคำสำคัญในกลุ่ม Hard Skill และ Soft Skill โดยไม่ได้รวมข้อมูลเชิงพฤติกรรมหรือบริบทอื่น ๆ เช่น ระดับประสบการณ์ หรือรายละเอียดบริษัท ซึ่งอาจช่วยเพิ่มความแม่นยำในการพยากรณ์

6) แม้ว่าจะใช้ LSTM ซึ่งเป็นเทคนิค Deep Learning ที่มีประสิทธิภาพสูงกับข้อมูลลำดับ (sequence data) แต่กลับให้ผลลัพธ์ต่ำที่สุดในชุดข้อมูลนี้ ซึ่งอาจสะท้อนถึงความไม่เหมาะสมกับลักษณะของข้อมูลที่แยกคำไว้แล้ว (structured TF-IDF / one-hot encoded)

เอกสารอ้างอิง

[1] De', R., Pandey, N., & Pal, A. (2020). Impact of digital surge during Covid-19 pandemic: A viewpoint on research and practice. International Journal of Information Management, 55, 102171.

[2] Lipaj, D., & Davidavičiene, V. (2013). Influence of Information Systems on business performance. Mokslas, 5(1), 38-45.

[3] Indeed Career Guide. (2020). Why Demand of Software Engineers is High. Retrieved from https://www.indeed.com/career-advice/finding-a-job/demand-of-software-engineers.

[4] P. Bourque and R. E. Fairley, "Guide to the Software Engineering Body of Knowledge (SWEBOK®): Version 3.0". IEEE Computer Society, 2014.

[5] IEEE Computer Society, "The Guide to the Software Engineering Body of Knowledge (SWEBOK® Guide V4.0)". IEEE, 2023.

[6] Varava, I., Bohinska, A., Vakaliuk, T., Mintii, I. (2021). Soft Skills in Software Engineering Technicians Education. Journal of Physics: Conference Series, 1946(1), 1–12. http://doi.org/10.1088/1742-6596/1946/1/012012.

[7] Charutwinyo, C. (2021). Competency Model of Software Developer in Thailand: A Qualitative Exploration. International Journal of Advance Trends in Computer Sciences and Engineering (IJATCSE), 10(3), 1524–1532. http://doi.org/10.30534/ijatcse/2021/061032021.

[8] Assyne, N., Ghanbari, H., Pulkkinen, M. (2022). The state of research on software engineering competencies: A systematic mapping study. Journal of Systems and Software, 185, 111183. https://doi.org/10.1016/j.jss.2021.111183.

[9] Assyne, N., Ghanbari, H., Pulkkinen, M. (2022). The essential competencies of software professionals: A unified competence framework. Information and Software Technology, 151, 107020. https://doi.org/10.1016/j.infsof.2022.107020.

เอกสารอ้างอิง

[10] Siddoo, V., Sawattawee, J., Janchai, W., Thinnukool, O. (2019). An exploratory study of digital workforce competency in Thailand. Heliyon, 5(5), e01723. https://doi.org/10.1016/j.heliyon.2019.e01723.

[11] Ternikov, A. (2022). Soft and hard skills identification: insights from IT job advertisements in the CIS region. PeerJ Comput Sci, 8, e946. https://doi.org/10.7717/peerj-cs.946.

[12] Matturro, G., Raschetti, F., & Fontán, C. (2019). A Systematic Mapping Study on Soft Skills in Software Engineering. Journal of Universal Computer Science, 25(1), 16-41. https://doi.org/10.21125/edulearn.2016.0939.

[13] Dada, O. A., Obaido, G., Sanusi, I. T., Aruleba, K., & Yunusa, A. A. (2023). Hidden Gold for IT Professionals, Educators, and Students: Insights From Stack Overflow Survey. IEEE Transactions on Computational Social Systems, 10(2), 795-806. doi: 10.1109/TCSS.2022.3151130.

[14] Hiranrat, C., & Harncharnchai, A. (2018). Using Text Mining to Discover Skills Demanded in Software Development Jobs in Thailand. Proceedings of the 2nd International Conference on Education and Multimedia Technology, Okinawa Japan, pp. 112-116. https://doi.org/10.1145/3206129.3239426.

[15] Hall, J. L., & Rao, A. (2020). Non-Technical skills needed by cyber security graduates. 2020 IEEE Global Engineering Education Conference (EDUCON), Porto, Portugal, pp. 354-358. doi: 10.1109/EDUCON45650.2020.9125105.

[16] Shadrin, D., Sozykin, A., Kuklin, E., & Bersenev, A. (2022). Classification of Software Developer Skills Using Analysis of Russian Job Ads. 2022 Ural-Siberian Conference on Biomedical Engineering, Radioelectronics and Information Technology (USBEREIT), Yekaterinburg, Russian Federation, pp. 236-239. doi: 10.1109/USBEREIT 56278.2022.9923408.

[17] Boyatzis, R. E. (1991). The competent manager: A model for effective performance. Wiley.https://www.researchgate.net/publication/247813294_The_Competent_Manager_A_Model_For_Effective_Performance

เอกสารอ้างอิง

[18] McClelland, D. C. (1973). Testing for competence rather than intelligence. American Psychologist, 28(1), 1–14. https://pubmed.ncbi.nlm.nih.gov/1746769/

[19] Parry, S. B. (1996). The quest for competencies. Training, 33(7), 48–56. https://eric.ed.gov/?id=EJ527012

[20] Hodkinson, P., & Issitt, M. (1995). The challenge of competence: Professionalism through vocational education and training. Cassell Education.

[21] Cheetham, G., & Chivers, G. (1998). The reflective (and competent) practitioner: A model of professional competence which seeks to harmonise the reflective practitioner and competence based approaches. Journal of European Industrial Training, 22(7), 267–276.

[22] Delamare Le Deist, F., & Winterton, J. (2005). What is competence? Human Resource Development International, 8(1), 27–46. https://www.tandfonline.com/doi/abs/10.1080/1367886042000338227

[23] Auld, E., & Morris, P. (2019). The OECD and IELS: Redefining early childhood education for the 21st century. Policy Futures in Education, 17(1), 11–26. https://discovery.ucl.ac.uk/id/eprint/10066928/7/Morris_The%20OECD%20and%20IELS.pdf?utm_source=chatgpt.com

[24] Patacsil, F. F., & Tablatin, C. L. S. (2017). Exploring the importance of soft and hard skills as perceived by IT internship students and industry: A gap analysis. Journal of Technology and Science Education, 7(3), 347–368.

[25] Zhang, A. (2012). Peer assessment of soft skills and hard skills. Journal of Information Technology Education: Research, 11(1), 155–168.

[26] Dixon, J., Belnap, C., Albrecht, C., & Lee, K. (2010). The importance of soft skills. Corporate Finance Review, 14(6), 35.

[27] Biggs, J. B. (1989). Approaches to the enhancement of tertiary teaching. Higher Education Research and Development, 8(1), 7–25.

[28] Skinner, B. F. (1965). Science and human behavior. Simon and Schuster.

เอกสารอ้างอิง

[29] Salman, M., Ganie, S. A., & Saleem, I. (2020). The concept of competence: A thematic review and discussion. European Journal of Training and Development.

[30] Le Deist, F. D., & Winterton, J. (2005). What Is Competence? Human Resource Development International, 8, 27-46. https://doi.org/10.1080/1367886042000338227

[31] IBM Software Business Analytics. (2012). CRISP-DM: Cross-industry standard process for data mining.

[32] IBM Corp. (2013). IBM SPSS Modeler CRISP-DM Guide (Version 18.3) [Documentation]. IBM. Retrieved from https://www.ibm.com/docs/en/spss-modeler/crisp-dm

[33] Salton, G., & Buckley, C. (1988). Term-weighting approaches in automatic text retrieval. Information Processing & Management, 24(5), 513–523.

[34] Ramos, J. (1999). Using TF-IDF to determine word relevance in document queries (Technical Report No. CS-99-06). University of Texas at Austin, Department of Computer Sciences.

[35] Yang, Y., & Pedersen, J. O. (1997). A comparative study on feature selection in text categorization. In Proceedings of the 14th International Conference on Machine Learning (pp. 412–420). Morgan Kaufmann Publishers.

[36] Antonie, M.-L., & Zaïane, O. R. (2002). Mammography classification by an association rule-based classifier. Proceedings of the International Conference on Machine Learning and Applications (ICMLA), 1, 801–806.

[37] Hu, M., & Liu, B. (2004). Mining and summarizing customer reviews. In Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (pp. 168–177).

[38] Brink, H., Richards, J., & Fetherolf, M. (2016). Real-world machine learning. Manning Publications.

เอกสารอ้างอิง

[39] Xuan, S., Liu, G., Li, Z., Zheng, L., Wang, S., & Jiang, C. (2018). Random forest for credit card fraud detection. 2018 IEEE 15th International Conference on Networking, Sensing and Control (ICNSC), Zhuhai, China, 27–29 March 2018, pp. 1–6.

[40] Wang, C., Deng, C., & Wang, S. (2020). Imbalance-XGBoost: Leveraging weighted and focal losses for binary label-imbalanced classification with XGBoost. Pattern Recognition Letters, 136, 190–197.

[41] Nguyen, H. Q., Nguyen, D. D. K., Le, T. D., Mai, A., & Huynh, K. T. (2023). Career path prediction using XGBoost model and students' academic results. CTU Journal of Innovation and Sustainable Development, 15(ISDS), 62–75. https://doi.org/10.22144/ctujoisd.2023.036

[42] Repaso, J. A. A., & Caparino, E. T. (2020). Analyzing and Predicting Career Specialization using Classification Techniques. International Journal of Advanced Trends in Computer Science and Engineering, 9(13), 342-348.

[43] Kamal, A., Naushad, B., Rafiq, H., & Tahzeeb, S. (2021). Smart Career Guidance System. 2021 4th International Conference on Computing & Information Sciences (ICCIS), Karachi, Pakistan, 1-7.

[44] Ali, I., Mughal, N., Khand, Z. H., Ahmed, J., & Mujtaba, G. (2022). Resume classification system using natural language processing and machine learning techniques. Mehran University Research Journal Of Engineering & Technology, 41(1), 65-79.

[45] Ortiz, G., Enguix, G. B., Gómez-Adorno, H., Ameer, I., & Sidorov, G. (2022). Job offers classifier using neural networks and oversampling methods. arXiv. https://arxiv.org/abs/2207.06223

[46] Chuangchai, S. and Boonbrahm, P. "Using text mining to discover skills demanded in software development jobs in Thailand," in Proceedings of the International Conference on Software Engineering, 2018.

เอกสารอ้างอิง

[47] Tanantong, T and Parnkow, M. (2022) “A Survey of Automatic Text Classification Based on Thai Social Media Data”. International Journal of Knowledge and Systems Science (IJKSS), 13(1), 1–25. https://doi.org/10.4018/IJKSS.312578

[48] Siddoo, V., Sawattawee, J., Janchai, W., Thinnukool, O. (2019) “An Exploratory Study Of Digital Workforce Competency In Thailand”, Heliyon, 5(5), e01723, ISSN 2405-8440, https://doi.org/10.1016/j.heliyon.2019.e01723.

[49] Meesad, P. (2012). “Thai Fake News Detection Based on Information Retrieval, Natural Language Processing and Machine Learning”. SN COMPUT. SCI. 2, 425 https://doi.org/10.1007/s42979-021-00775-6