การคัดกรองผู้สมัครจากประวัติย่อด้วยหลักการเรียนรู้ของเครื่อง

CANDIDATE SCREENING FROM RESUME USING MACHINE LEARNING

นงนุช เสถียรกิตติโรจน์

บัณฑิตวิทยาลัย มหาวิทยาลัยศรีนครินทรวิโรฒ

2566

การคัดกรองผู้สมัครจากประวัติย่อด้วยหลักการเรียนรู้ของเครื่อง

นงนุช เสถียรกิตติโรจน์

สารนิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
วิทยาศาสตรมหาบัณฑิต สาขาวิชาวิทยาการข้อมูล
คณะวิทยาศาสตร์ มหาวิทยาลัยศรีนครินทรวิโรฒ
ปีการศึกษา 2566

CANDIDATE SCREENING FROM RESUME USING MACHINE LEARNING

NONGNUCH SATHEANKITTIROJ

A Master's Project Submitted in Partial Fulfillment of the Requirements

for the Degree of MASTER OF SCIENCE

(Data Science)

Faculty of Science, Srinakharinwirot University

2023

สารนิพนธ์

เรื่อง

การคัดกรองผู้สมัครจากประวัติย่อด้วยหลักการเรียนรู้ของเครื่อง

ของ

นงนุช เสถียรกิตติโรจน์

ได้รับอนุมัติจากบัณฑิตวิทยาลัยให้นับเป็นส่วนหนึ่งของการศึกษาตามหลักสูตร

ปริญญาวิทยาศาสตรมหาบัณฑิต สาขาวิชาวิทยาการข้อมูล

ของมหาวิทยาลัยศรีนครินทรวิโรฒ

(รองศาสตราจารย์ นายแพทย์ฉัตรชัย เอกปัญญาสกุล)

คณบดีบัณฑิตวิทยาลัย

คณะกรรมการสอบปากเปล่าสารนิพนธ์

..... ที่ปรึกษาหลัก

(อาจารย์ ดร.วีระ สอิ้ง)

..... ประธาน

(รองศาสตราจารย์ ดร.สุพัฒนา เอื้อทวีเกียรติ)

..... กรรมการ

(อาจารย์ ดร.ศุภร คนธภักดี)

ชื่อเรื่อง	การคัดกรองผู้สมัครจากประวัติย่อด้วยหลักการเรียนรู้ของเครื่อง
ผู้วิจัย	นงนุช เสถียรกิตติโรจน์
ปริญญา	วิทยาศาสตรมหาบัณฑิต
ปีการศึกษา	2566
อาจารย์ที่ปรึกษา	อาจารย์ ดร. วีระ สอิ้ง

บุคลากรถือเป็นทรัพยากรที่มีความสำคัญมากที่สุดขององค์กร ดังนั้นการสรรหาบุคลากรที่มีทักษะความสามารถที่เหมาะสมกับตำแหน่งงานจึงมีความสำคัญอย่างยิ่ง ในปัจจุบันผู้คัดสรรจะทำการพิจารณาผู้สมัครงานด้วยวิธีการอ่านข้อมูลจากในประวัติย่อ จากประสบการณ์ของมนุษย์เป็นหลัก จึงทำให้เกิดความผิดพลาดและความล่าช้าขึ้น เพื่อเพิ่มประสิทธิภาพและความรวดเร็วของการคัดสรรจึงทำให้เกิดงานวิจัยนี้ขึ้น โดยวัตถุประสงค์เพื่อศึกษาโครงสร้างแบบจำลองการคัดกรองผู้สมัครจากประวัติย่อด้วยหลักการเรียนรู้ของเครื่องประเภทการเรียนรู้แบบมีผู้สอน ซึ่งเป็นกระบวนการทำงานของการประมวลผลภาษาธรรมชาติ งานวิจัยนี้ทำการทดลองและเปรียบเทียบประสิทธิภาพแบบจำลอง 8 แบบ ได้แก่ Support Vector Classification (SVC), Logistic Regression, Random Forest, K-Nearest Neighbors (KNN), Gradient Boosting, AdaBoost Classifier, Gaussian Naïve Bayes และ Decision Tree โดยใช้ข้อมูลจากแหล่งข้อมูลสาธารณะ ซึ่งในชุดข้อมูลประกอบด้วยประเภทงานและประวัติย่อ ขนาด 962 แถว 2 คอลัมน์ จากผลการทดลองพบว่าแบบจำลอง Support Vector Classification (SVC), Logistic Regression และ Random Forest มีค่าความถูกต้องสูงที่สุดเท่ากัน 99.48% แต่เมื่อวิเคราะห์ค่า Cross Validation ประกอบพบว่าแบบจำลอง Support Vector Classification (SVC) มีค่ามากที่สุดอยู่ที่ 99.50% และมีค่า Precision 99.50%, Recall 99.71%, F1 Score 99.58% อีกทั้งยังได้ทำการหาคุณลักษณะที่มีความสำคัญต่อการจำแนกประเภทของสายงานจากประวัติย่อโดยใช้วิธีการคำนวณค่า Coefficients และค่า SHAP Value ซึ่งจากผลลัพธ์ของคุณลักษณะเหล่านี้ จะช่วยให้เข้าใจถึงปัจจัยที่ส่งผลต่อการจำแนกประเภทสายงานได้ดียิ่งขึ้น และสามารถนำไปปรับปรุงและพัฒนาโมเดลต่อไปในอนาคตได้ กล่าวโดยสรุปคือแบบจำลองที่มีประสิทธิภาพดีที่สุดและเหมาะสมที่สุดในการช่วยคัดกรองผู้สมัครจากประวัติย่อได้แก่แบบจำลอง SVC

คำสำคัญ : การคัดกรองประวัติย่อ, การเรียนรู้ของเครื่อง, การประมวลผลภาษาธรรมชาติ, การจำแนกประเภทหลายหมวดหมู่

Title	CANDIDATE SCREENING FROM RESUME USING MACHINE LEARNING
Author	NONGNUCH SATHEANKITTIROJ
Degree	MASTER OF SCIENCE
Academic Year	2023
Thesis Advisor	Vera Sa-Ing , Ph.D.

Staff are considered to be the most important resource in an organization. Therefore, recruiting staff have crucial skills and abilities to identify the best fit for a job position is. Currently, selectors often evaluate job applicants by reading information from resumes, primarily using human judgment, leading to errors and delays. This research was conducted to enhance the efficiency and speed of the selection process. The objective was to study and construct a model for screening job applicants from resumes using machine learning principles, and specifically supervised learning. The workflow included natural language processing experimentation and comparison of the performance of eight models: Support Vector Classification (SVC), Logistic Regression, Random Forest, K-Nearest Neighbors (KNN), Gradient Boosting, AdaBoost Classifier, Gaussian Naïve Bayes, and Decision Tree. Publicly available data, consisting of job categories and resumes, were utilized. With a dataset size of 962 rows and two columns, experimental results indicated that the Support Vector Classification (SVC), Logistic Regression, and Random Forest models achieved the highest accuracy of 99.48%. Cross-validation revealed that the Support Vector Classification (SVC) model performed the best with an accuracy of 99.50%, Precision of 99.50%, Recall of 99.71%, and F1 Score of 99.58%. Furthermore, significant features for job category classification from resumes were identified using Coefficients and SHAP Values. These features facilitate better understanding of factors influencing job category classification, aiding in model improvement and development for future use. In summary, the most efficient and suitable model for screening job applicants from resumes was found to be the SVC model.

Keyword : Resume Screening, Machine Learning, Natural Language Processing, Multi-Class Classification

กิตติกรรมประกาศ

รายงานสารนิพนธ์ฉบับนี้สำเร็จสมบูรณ์ไปได้ เนื่องจากได้รับความช่วยเหลือและความเอาใจใส่อย่างยิ่งจาก อาจารย์ ดร.วีระ สอิ้ง อาจารย์ประจำภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยศรีนครินทรวิโรฒ และที่ปรึกษาหลักในงานวิจัยครั้งนี้ ที่ได้เสียสละเวลาส่วนตัวมาให้ความรู้ ให้คำปรึกษาในการทำงานเสมอมา ตลอดจนช่วยแนะนำและแก้ไขข้อบกพร่องต่าง ๆ ในการทำงานวิจัยครั้งนี้ จนกระทั่งงานวิจัยฉบับนี้สำเร็จลุล่วงไปได้ด้วยดี ขอกราบขอบพระคุณอย่างสูง มา ณ ที่นี้

ขอขอบพระคุณ รองศาสตราจารย์ ดร.สุพัฒนา เอื้อทวีเกียรติ ประธานผู้ทรงคุณวุฒิ ในการสอบปากเปล่าสารนิพนธ์ที่กรุณาให้คำแนะนำที่เกี่ยวกับงานวิจัย เพื่อนำความรู้ที่ได้ไปปรับใช้ในอนาคต

ขอขอบพระคุณคณาจารย์ทุกท่านและกรรมการบริหารหลักสูตร สาขาวิทยาการข้อมูล คณะวิทยาศาสตร์ มหาวิทยาลัยศรีนครินทรวิโรฒ ที่ให้ความรู้ ประสบการณ์อันเป็นประโยชน์ต่อการศึกษา และการทำงานวิจัยนี้

ขอขอบพระคุณเพื่อน ๆ นิสิตปริญญาโทร่วมชั้นเรียนทุกท่าน ที่คอยช่วยเหลือ สนับสนุนและเป็นกำลังใจให้ผู้วิจัยมาโดยตลอด

ขอขอบพระคุณครอบครัว และสิ่งศักดิ์สิทธิ์ทั่วสากลพิภพที่เป็นกำลังใจ คอยดูแลให้การเรียน การสอบ การทำงานวิจัยลุล่วงอย่างราบรื่น

สุดท้ายนี้ผู้วิจัยหวังเป็นอย่างยิ่งว่า งานวิจัยฉบับนี้จะเป็นประโยชน์แก่ผู้ที่เกี่ยวข้องและสนใจต่อไปไม่มากก็น้อย หากมีข้อผิดพลาดประการใด ผู้วิจัยขอน้อมรับและขออภัยมา ณ ที่นี้ด้วย

นงนุช เสถียรกิตติโรจน์

สารบัญ

หน้า

สารบัญตาราง

หน้า

สารบัญรูปภาพ

หน้า

บทที่ 1
บทนำ

1.1 ที่มาและความสำคัญของงานวิจัย

กระบวนการสรรหาบุคลากร (Recruitment Process) หมายถึง กระบวนการในการค้นหาและคัดกรองบุคลากรที่มีคุณสมบัติเหมาะสมกับตำแหน่งงานขององค์กร เพื่อคัดเลือกเข้ามาปฏิบัติงานในตำแหน่งตามที่องค์กรต้องการ

การสรรหาบุคลากรนั้นมีความสำคัญต่อองค์กรเป็นอย่างมาก เนื่องจากบุคลากรถือเป็นทรัพยากรที่มีความสำคัญมากที่สุดขององค์กร ซึ่งบุคลากรไม่ว่าจะตำแหน่งใดก็ตามเปรียบเสมือนฟันเฟืองที่มีความสำคัญไม่แพ้กัน ถ้าฟันเฟืองหรือตำแหน่งใดตำแหน่งหนึ่งขาดไปอาจทำให้องค์กรประสบปัญหาไม่สามารถบรรลุเป้าหมายและพบกับความสำเร็จได้ ในทางตรงกันข้าม ถ้าหากองค์กรใดมีบุคลากรที่ทำงานได้อย่างมีประสิทธิภาพครบทุกตำแหน่ง องค์กรนั้นจะก้าวไปสู่ความสำเร็จได้อย่างไม่ยาก ดังนั้นก่อนที่จะมีบุคลากรที่มีความสามารถอยู่ในองค์กรได้นั้น ต้องอาศัยกระบวนการสรรหาบุคลากรที่มีประสิทธิภาพ

กระบวนการสรรหาบุคลากรสามารถแบ่งออกเป็น 7 ขั้นตอน ดังนี้

1. วางแผนการสรรหาและคัดเลือก (Planning and Recruitment) ขั้นตอนนี้ถือเป็นขั้นตอนเริ่มต้นของการสรรหา โดยต้องกำหนดแผนการทำงานทุกขั้นตอน โดยพิจารณาจากความต้องการขององค์กร ตำแหน่งที่ต้องการ คุณสมบัติของผู้สมัคร งบประมาณ และระยะเวลาในการสรรหาเพื่อจัดจ้างบุคลากรได้ทันกับความต้องการขององค์กร

2. กำหนดรายละเอียด (Job Description) ของงานและคุณสมบัติ (Qualification) ของผู้สมัคร ขั้นตอนของการจัดทำรายละเอียดลักษณะงาน คุณสมบัติและความสามารถของผู้สมัคร หน้าที่และความรับผิดชอบของงาน รวมไปถึงผลประโยชน์และสวัสดิการต่าง ๆ ที่พนักงานจะได้รับ

3. สื่อสารการรับสมัครงาน (Communication) การเผยแพร่ข้อมูลการรับสมัครงานไปยังผู้สมัครงานที่ตรงกับเป้าหมายในแต่ละตำแหน่งงาน โดยช่องทางการเปิดรับสมัครงานมีได้หลายวิธี เช่น ติดป้ายประกาศหน้าบริษัทหรือโรงงาน ประกาศงานลงบนสื่อสังคมออนไลน์ต่าง ๆ ประกาศงานผ่านเว็บไซต์หรือสิ่งพิมพ์ เป็นต้น ซึ่งช่องทางที่ได้รับความนิยมในยุคนี้และมีประสิทธิภาพมากคือ การประกาศงานผ่านเว็บไซต์จัดหางานชั้นนำของประเทศ

4. การคัดสรรบุคลากร (Selection)เมื่อมีผู้สมัครส่งใบสมัครงานเข้ามาแล้ว ก็มาถึงขั้นตอนที่สำคัญคือขั้นตอนการคัดกรองผู้สมัครงานโดยผู้คัดสรรจะทำการพิจารณาจาก

คุณสมบัติ ทักษะความสามารถ ประสบการณ์ของผู้สมัครที่สอดคล้องเกี่ยวข้องกับตำแหน่งงานนั้น ๆ วิธีการคัดกรองแบ่งออกเป็น 2 วิธีคือ

4.1 คัดกรองด้วยสายตามนุษย์ โดยแบ่งออกเป็น 2 รอบ ในรอบแรกจะมองหาเฉพาะทักษะเฉพาะทาง (Hard skill) ที่จำเป็นกับตำแหน่งงาน และรอบสอง อ่านรายละเอียดทั้งหมดในประวัติย่อ (Resume) ของผู้สมัคร

4.2 คัดกรองแบบอัตโนมัติด้วยหลักการเรียนรู้ของเครื่อง (Machine Learning) ซึ่งวิธีนี้เป็นวิธีที่ผู้วิจัยสนใจ และทำการศึกษาเพิ่มเติมจนเกิดเป็นงานวิจัยนี้

5. สัมภาษณ์งาน (Job Interview) ขั้นตอนนี้จะทำให้ทั้งผู้สัมภาษณ์และผู้ถูกสัมภาษณ์ได้พบกันโดยจะผ่านช่องทางออนไลน์หรือมาสัมภาษณ์ที่บริษัทก็สุดแล้วแต่ความสะดวก ซึ่งผู้สัมภาษณ์งานมักจะถามคำถามเพื่อให้ได้ทราบถึงความสามารถ คุณสมบัติและทักษะที่แท้จริงของผู้สมัครงาน

6. เซ็นสัญญาจ้างงาน (Signing of Employment Contract) หลังจากสัมภาษณ์งานเรียบร้อยแล้ว นายจ้างจะตัดสินใจเลือกผู้สมัครเข้าทำงาน โดยพิจารณาจากคุณสมบัติต่าง ๆ จากนั้นเมื่อได้ผู้สมัครที่ผ่านการคัดเลือกแล้ว จะมีการนัดเซ็นสัญญาจ้างงาน โดยผู้สรรหาต้องชี้แจงรายละเอียดเกี่ยวกับข้อกำหนด อัตราค่าจ้าง สวัสดิการ และผลตอบแทนอื่น ๆ อย่างชัดเจน และเริ่มปฏิบัติงานจริงได้ โดยคุณสมบัติทั่วไปมีดังนี้

6.1 ทักษะเฉพาะทางที่จำเป็นในการทำงาน (Hard Skill) เช่น ทักษะด้านเทคนิค ทักษะการใช้เครื่องมือ ทักษะในการทำงานเฉพาะสาขาวิชา เป็นต้น

6.2 ทักษะส่วนบุคคลที่จำเป็นในการทำงาน (Soft Skill) เช่น ทักษะการสื่อสาร ทักษะการทำงานเป็นทีม ทักษะการแก้ปัญหา เป็นต้น

6.3 ประสบการณ์การทำงาน (Experience) เป็นประสบการณ์ที่เกี่ยวข้องกับตำแหน่งงานนั้น ๆ ทั้งทางตรง และทางอ้อม เช่น ประสบการณ์การฝึกงาน ประสบการณ์จากการทำกิจกรรมนอกหลักสูตร ประสบการณ์การทำงานประจำ หรืองานชั่วคราว เป็นต้น

6.4 ความเหมาะสมของพนักงานกับวัฒนธรรมองค์กร (Culture Fit) โดยผู้สมัครมีลักษณะนิสัย ค่านิยม และทัศนคติที่สอดคล้องกับวัฒนธรรมองค์กร

7. ประเมินผลการสรรหา (Evaluation of Recruitment) หลังจากได้บุคลากรที่มีความสามารถตรงกับที่องค์กรต้องการแล้ว ผู้สรรหาต้องกลับมาทบทวนและสรุปขั้นตอนของกระบวนการสรรหาเพื่อหาแนวทางการปรับปรุงให้มีประสิทธิภาพที่ดีขึ้น โดยอาจพิจารณาจาก

รายละเอียดในประกาศงาน ช่องการการประกาศงาน จำนวนผู้สมัครงานที่ได้รับ ระยะเวลาในการสรรหา งบประมาณที่ใช้ และผลลัพธ์ของการสรรหา (HREX.asia, 2562)

จะเห็นว่ากระบวนการสรรหาบุคลากรนั้นใช้เวลาค่อนข้างมาก โดยเฉพาะขั้นตอนการคัดกรองผู้สมัครงานจากประวัติย่อ (Resume) ที่ผู้คัดสรรจะต้องใช้สายตาอ่านใบสมัคร และคัดเลือกผู้ที่มีความสามารถตรงกับที่องค์กรต้องการมากที่สุดในระยะเวลาที่จำกัด นั่นจึงเป็นความท้าทายอย่างมากของผู้คัดสรร ยิ่งไปกว่านั้นถ้าผู้คัดสรรอยู่ในองค์กรที่มีชื่อเสียงและได้รับความนิยมจากผู้สมัครงานมาก อาจมีใบสมัครเข้ามากว่า 1,000 ใบสมัครต่อหนึ่งวันต่อหนึ่งตำแหน่งงานก็เป็นได้ ถ้าใช้สายตามนุษย์ในการคัดกรองอย่างเดียวนั้น อาจเกิดความผิดพลาดได้ง่าย (Human error) และใช้ระยะเวลานาน

ดังนั้นองค์กรชั้นนำต่าง ๆ โดยเฉพาะบริษัทจัดหางาน (Recruitment Agency) รวมถึงผู้วิจัยจึงมีความคิดที่จะนำเทคโนโลยีปัญญาประดิษฐ์เข้ามาช่วยคัดกรองผู้สมัครแบบอัตโนมัติ (Automation) เพื่อลดความผิดพลาดของมนุษย์ (Human error) และลดระยะเวลาในการทำงานลง ซึ่งจากการศึกษางานวิจัยในอดีตและปัจจุบันพบว่าวิธีการคัดกรองผู้สมัครจากประวัติย่อด้วยหลักการการเรียนรู้ของเครื่อง (Machine Learning) และการประมวลผลภาษาธรรมชาติ (Natural Language Processing) สามารถช่วยคัดกรองผู้สมัครจากทักษะ การศึกษา ประสบการณ์การทำงานของแต่ละสายงานได้ในเวลาไม่กี่นาที จึงช่วยลดเวลาในการทำงาน และช่วยลดความผิดพลาดที่เกิดจากมนุษย์ได้ตามที่ต้องการ

1.2 วัตถุประสงค์ของงานวิจัย

เพื่อศึกษาและเปรียบเทียบประสิทธิภาพแบบจำลองต่าง ๆ ของการเรียนรู้ของเครื่อง (Machine Learning) ประเภทการเรียนรู้แบบมีผู้สอน (Supervised Machine Learning) ซึ่งเป็นแบบจำลองที่ใช้ในการจำแนกประเภท (Classification) เช่น Support Vector Classification (SVC), Logistic Regression, Random Forest, K-Nearest Neighbors, Gradient Boosting, AdaBoost Classifier, Gaussian Naïve Bayes, Decision Tree และเทคนิค One Vs Rest Classifier

เพื่อศึกษาการทำงานของการประมวลผลภาษาธรรมชาติ (Natural Language Processing) เช่น การกำหนดประเภทของคำในประโยค (Parts of Speech), การตัดคำ (Tokenization), การแปลงข้อความเป็นเวกเตอร์ตัวเลข (TF-IDF Vectorization) เป็นต้น

เพื่อค้นหาเครื่องมือที่มีประสิทธิภาพในการคัดกรองผู้สมัครจากประวัติย่อแบบอัตโนมัติ เพื่อลดเวลาการทำงานของผู้สรรหา และลดความผิดพลาดที่เกิดจากการคัดกรองด้วยสายตามนุษย์

1.3 ขอบเขตของงานวิจัย

1. ข้อมูลที่ใช้ในงานวิจัยฉบับนี้มาจาก Public Dataset ชื่อ Updated Resume Dataset

2. ข้อมูลประวัติย่อ (Resume) ที่ใช้ในงานวิจัยนี้เป็นภาษาอังกฤษ และส่วนใหญ่อยู่ในสายงานไอที

3. ข้อมูลที่ใช้คัดกรองผู้สมัครจากประวัติย่อส่วนใหญ่มาจากทักษะ การศึกษา และประสบการณ์ของผู้สมัครงาน

4. ประเมินประสิทธิภาพการเรียนรู้แบบมีผู้สอน (Supervised Machine Learning) ของแบบจำลองการจำแนกแบบไบนารี (Binary Classification) เช่น Support Vector Classification (SVC), Logistic Regression, Random Forest, K-Nearest Neighbors, Gradient Boosting, AdaBoost Classifier, Gaussian Naïve Bayes, Decision Tree รวมทั้งการจำแนกประเภทหลายคลาส (Multi-Class Classification) โดยใช้กลยุทธ์แบบ One-vs-Rest

1.4 ขั้นตอนของการทำงานวิจัย

ศึกษาการวิเคราะห์ข้อความของชุดข้อมูลประวัติย่อด้วยการประมวลผลภาษาธรรมชาติ หรือ Text Analytics of Resume Dataset with NLP (Majumder, 2022)

ศึกษางานวิจัยเรื่องการคัดกรองประวัติย่อด้วยหลักการเรียนรู้ของเครื่อง เพื่อทำงานวิจัยต่อยอดโดยการปรับปรุงการใช้เทคนิคการเรียนรู้ของเครื่อง และประเมินประสิทธิภาพของแบบจำลอง

การนำเข้าข้อมูล การเตรียมข้อมูล การแปลงข้อมูล การจัดการกับคำในประโยค เช่น การตัดคำ (Tokenization), การลบช่องว่าง, การลดรูปแบบคำลงเหลือรูปแบบพื้นฐาน (Stemming and Lemmatization), การกำหนดประเภทของคำ (Parts of Speech), สร้างคลังคำศัพท์และแปลงข้อความเป็นเวกเตอร์ตัวเลข (Bag of Words), ไฮไลท์คำที่โดดเด่นออกมา (TF-IDF Vectorization)

การวิเคราะห์ข้อมูลเชิงสำรวจพบว่ามีข้อมูลประวัติย่อทั้งหมด 962 ประวัติย่อ และมีสายงานทั้งหมด 25 สายงาน โดยสายงานที่มีประวัติย่อมากที่สุด 3 อันดับแรกได้แก่ Java developer, Testing และ DevOps Engineer

สร้างแบบจำลองการคัดกรองผู้สมัครจากประวัติย่อด้วยหลักการเรียนรู้ของเครื่องแบบต่าง ๆ ตามที่ได้ศึกษาในงานวิจัยประกอบด้วย OneVsRestClassifier ที่ใช้สำหรับปัญหาการจำแนกประเภทหลายคลาส (multi-class classification), Support Vector Classification (SVC), Logistic Regression, Random Forest, K-Nearest Neighbors, Gradient Boosting, AdaBoost Classifier, Gaussian Naïve Bayes, Decision Tree

ประเมินประสิทธิภาพของแบบจำลองการเรียนรู้ของเครื่องแบบต่าง ๆ ที่ใช้ในการทดลองการคัดกรองผู้สมัครจากประวัติย่อ

วิเคราะห์และอภิปรายผลการทดลองของแบบจำลองการคัดกรองผู้สมัครจากประวัติย่อและเสนอแนะข้อจำกัดและการปรับปรุง

1.5 ประโยชน์ที่คาดว่าจะได้รับจากงานวิจัย

1. ได้รับความรู้ความเข้าใจเกี่ยวกับการคัดกรองผู้สมัครจากประวัติย่อด้วยหลักการเรียนรู้ของเครื่องแบบต่าง ๆ
2. สร้างแบบจำลองต่าง ๆ พร้อมทั้งสามารถประเมินประสิทธิภาพของแบบจำลองได้
3. ช่วยลดเวลาการทำงานของผู้สรรหาได้อย่างมาก
4. ช่วยลดความผิดพลาดที่อาจเกิดจากมนุษย์ในการอ่านประวัติย่อ
5. ช่วยลดอัตราการว่างงาน เมื่อการคัดสรรเป็นไปอย่างมีประสิทธิภาพและรวดเร็วขึ้น

บทที่ 2
ทบทวนวรรณกรรม และงานวิจัยที่เกี่ยวข้อง

ในงานวิจัยนี้ ผู้วิจัยได้ทำการศึกษาเอกสารและงานวิจัยที่เกี่ยวข้อง และได้นำเสนอตามหัวข้อต่อไปนี้

1. วิวัฒนาการของกระบวนการจ้างงาน
2. ทฤษฎีการประมวลผลภาษาธรรมชาติ (Natural Language Processing)
3. ทฤษฎีการวิเคราะห์ข้อมูลด้วยหลักการทางคอมพิวเตอร์
4. งานวิจัยที่เกี่ยวข้อง

2.1 วิวัฒนาการของกระบวนการจ้างงาน

2.1.1 ในยุคแรกของการจ้างงาน

นิยมใช้วิธีติดป้ายประกาศไว้หน้าบริษัท ลงโฆษณาในหนังสือพิมพ์ วารสาร หรือบอกกันปากต่อปาก ผู้ที่สนใจสมัครงานจะยื่นใบสมัครและประวัติในรูปแบบของกระดาษ หรือส่งเอกสารใบสมัครทางไปรษณีย์ไปยังบริษัท ทางทีมฝ่ายสรรหาจะต้องจัดเรียงใบสมัครและคัดเลือกผู้สมัครเข้ารับการสัมภาษณ์เพิ่มเติม กระบวนการทั้งหมดจะใช้เวลาและความพยายามของมนุษย์เป็นอย่างมากในการหาผู้สมัครที่เหมาะสมกับตำแหน่งงาน อีกทั้งใบสมัครที่ส่งทางไปรษณีย์อาจเสียหาย และสูญหายได้

2.1.2 ในยุคที่สองของการจ้างงาน

เมื่ออุตสาหกรรมต่าง ๆ เติบโตขึ้น ความต้องการในการจ้างงานก็เพิ่มขึ้นอย่างรวดเร็ว เพื่อตอบสนองความต้องการด้านการจ้างงานเหล่านี้ จึงมีเว็บไซต์จัดหางานเกิดขึ้นมากมาย ให้ผู้สมัครงานอัปโหลดประวัติย่อของตนเองในรูปแบบเฉพาะที่เว็บไซต์นั้นกำหนด จากนั้นค้นหาตำแหน่งงานที่ต้องการและส่งใบสมัครผ่านทางออนไลน์ไปยังบริษัทที่ต้องการจะสมัคร จะเห็นว่าระบบการอัปโหลดประวัติย่อเหล่านี้ไม่มีความยืดหยุ่น และยังคงต้องใช้สายตามนุษย์ในการคัดเลือกผู้สมัครอยู่

2.1.3 ในยุคที่สามของการจ้างงาน

การหางาน สมัครงานผ่านช่องทางออนไลน์ยังคงได้รับความนิยมในยุคนี้ และยังอนุญาตให้ผู้สมัครอัปโหลดประวัติย่อของตนเองในรูปแบบที่ยืดหยุ่นขึ้นกว่ายุคก่อน อีกทั้งยังนำเทคโนโลยีปัญญาประดิษฐ์เข้ามาช่วยวิเคราะห์และคัดกรองผู้สมัครด้วยการทำงานของการประมวลผลภาษาธรรมชาติ (Natural Language Processing) เพื่อเพิ่มประสิทธิภาพของการคัดสรร และช่วยลดเวลาในการทำงานลงได้ (Zubeda et al., 2015)

2.2. ทฤษฎีการประมวลผลภาษาธรรมชาติ (Natural Language Processing)

การประมวลผลภาษาธรรมชาติ เป็นการทำให้คอมพิวเตอร์เข้าใจภาษามนุษย์ รวมถึงไปการวิเคราะห์ทางด้านภาษาศาสตร์ การตีความจากข้อความ (Kullawattana, 2562) (datawow, 2563) (Bhavsar, 2023) (spaCy, 2023) (Daroontham, 2561b)

คำศัพท์และความหมายที่เกี่ยวข้องกับการประมวลผลภาษาธรรมชาติ

1. Tokenization กระบวนการแบ่งข้อความขนาดใหญ่เป็นคำย่อย ๆ ที่เรียกว่า tokens สิ่งนี้ทำได้โดยการลบหรือแยกตัวอักษร ช่องว่าง และเครื่องหมายวรรคออก สามารถทำได้หลากหลายวิธี โดยใช้ Natural Language Toolkit [NLTK] หรือไลบรารี spaCy เป็นต้น ขั้นตอนนี้ถือเป็นขั้นตอนที่จำเป็นสำหรับการประมวลผลข้อความในภายหลัง เช่น การลบคำหยุด การทำ Stemming และการทำ Lemmatization

2. Corpus หรือ Corpora คือ เอกสารตัวอย่างที่รวบรวมไว้ เพื่อเอาไว้เป็นข้อมูลในการอนุมาน และตรวจสอบความสมเหตุสมผลของกฎเกณฑ์ทางภาษาศาสตร์ และการวิเคราะห์ทางสถิติ (Kanoktipsatharporn, 2566)

3. Stop_words การกำจัดคำที่ไม่สำคัญ คำที่ปรากฏอยู่เยอะ เช่น คำเชื่อม

4. Stemming และ Lemmatization เป็นวิธีในการลดรูปแบบคำลงเหลือรูปแบบพื้นฐาน เทคนิคทั้งสองนี้ใช้สำหรับการประมวลผลภาษาธรรมชาติ (NLP) เพื่อทำให้การวิเคราะห์ข้อความง่ายขึ้น ซึ่ง Stemming คือกระบวนการกำจัดส่วนต่อท้ายของคำ เช่น -ing, -ed, -s, และ -es เพื่อสร้างรากของคำ ตัวอย่างเช่น "running" จะกลายเป็น "run" และ "loved" จะกลายเป็น "love" อัลกอริธึมที่ใช้ทั่วไปได้แก่

- Porter-Stemmer เป็นอัลกอริธึม stemming พื้นฐานที่พัฒนาโดย Martin Porter ในปี 1979 อัลกอริธึมนี้ใช้กฎเกณฑ์ง่าย ๆ เพื่อกำจัดส่วนต่อท้ายของคำ เช่น Porter-Stemmer จะเปลี่ยน "running" เป็น "run" และ "loved" เป็น "love"

- Snowball stemmer เป็นอัลกอริทึม stemming ที่พัฒนาโดย Martin Porter ในปี 1997 อัลกอริทึมนี้ใช้กฎเกณฑ์ที่ซับซ้อนกว่า Porter-Stemmer เพื่อให้ได้ผลลัพธ์ที่แม่นยำยิ่งขึ้น ตัวอย่างเช่น Snowball stemmer จะเปลี่ยน "running" เป็น "run" และ "loved" เป็น "love" เช่นเดียวกับ Porter-Stemmer แต่ Snowball stemmer จะเปลี่ยน "better" เป็น "good" และ "worst" เป็น "bad"

- Lancaster stemmer เป็นอัลกอริทึม stemming ที่พัฒนาโดย Geoffrey Lancaster ในปี 1980 อัลกอริทึมนี้ใช้กฎเกณฑ์ที่คล้ายกับ Porter-Stemmer แต่ Lancaster

stemmer จะลบส่วนต่อท้ายของคำได้มากกว่า Porter-Stemmer ตัวอย่างเช่น Lancaster stemmer จะเปลี่ยน "running" เป็น "run" และ "loved" เป็น "love" เช่นเดียวกับ Porter-Stemmer แต่ Lancaster stemmer จะเปลี่ยน "better" เป็น "good" และ "worst" เป็น "bad"

ส่วน Lemmatization การเปลี่ยนรูปคำให้อยู่ในรูปแบบของคำดั้งเดิมหรือคำกริยาช่องที่ 1 เพื่อให้อยู่ในรากศัพท์เดียวกัน เช่น is, am, are, was เปลี่ยนเป็น be และ saw, seen เปลี่ยนเป็น see

ความแตกต่างที่สำคัญระหว่าง Stemming และ Lemmatization คือ stemming เป็นการประมาณที่เร็วและง่ายกว่า ในขณะที่ lemmatization แม่นยำกว่า แต่ช้ากว่า สามารถดาวน์โหลดชุดข้อมูลและแบบจำลองที่ได้รับการฝึกฝนสำหรับ Natural Language Toolkit (NLTK) ซึ่งเป็นไลบรารี Python สำหรับประมวลผลภาษาธรรมชาติ ด้วยคำสั่ง nltk.download('wordnet') จะดาวน์โหลดชุดข้อมูล WordNet ซึ่งเป็นพจนานุกรมคำศัพท์ภาษาอังกฤษเชิงความหมาย WordNet แบ่งคำศัพท์ออกเป็นคำพ้องความหมาย คำพ้องรูป คำพ้องเสียง และคำที่เกี่ยวข้องอื่น ๆ ข้อมูลนี้สามารถใช้สำหรับการประมวลผลภาษาธรรมชาติหลายประเภท เช่น การจำแนกประเภทคำ การค้นหาคำพ้องความหมาย และการเชื่อมคำ

ตัวอย่าง

ข้อความต้นฉบับ: "The dog is running."

Stemming:

"dog" -> "dog"

"is" -> "is"

"running" -> "run"

Lemmatization:

"dog" -> "dog"

"is" -> "be"

"running" -> "run"

5. Parsing คือ กระบวนการในการระบุโครงสร้างของข้อความ โดยการวิเคราะห์คำที่เป็นส่วนประกอบ ด้วยหลักไวยากรณ์ของภาษา ผลลัพธ์ที่ได้ออกมาจะเป็นโครงสร้างแบบต้นไม้ เรียกว่า Parse Tree (Kanoktipsatharporn, 2566)

6. Word2Vec คือการสร้างแบบจำลองจาก Word embedding ที่ได้รับความนิยมมากที่สุด เปิดตัวในปี 2013 โดย Tomas Mikolov จาก Google AI โดยการนำ Word embedding หลาย ๆ ชั้นมาสร้างเป็นแบบจำลองซึ่งทำการ Training โดยการคำนวณตัวเลขจากบริเวณใกล้เคียง มี 2 ประเภทได้แก่

- CBOW (Continuous Bag of Words) จะพิจารณาบริบทของคำโดยดูจากคำที่อยู่รอบ ๆ คำนั้น

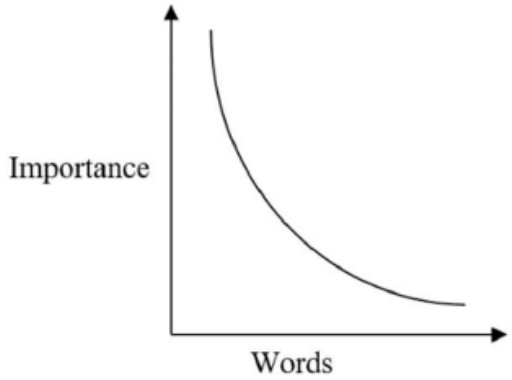
- Skip-gram จะพิจารณาบริบทของคำโดยดูจากคำที่คำนั้นสามารถแทนที่ได้

Term Frequency (TF) คือ การแสดงความถี่ของคำต่าง ๆ ที่ปรากฏในเอกสาร จำนวนครั้งที่ปรากฏของคำบ่งบอกถึงความหมายและสำคัญของคำนั้น ๆ ต่อเอกสาร โดยรวม Term Frequency มักถูกอ้างถึงบ่อยครั้งในบริบทของ Inverse Document Frequency (IDF) TF-IDF Vectorization คือเทคนิคการแปลงข้อความเป็นเวกเตอร์ตัวเลข ซึ่งสามารถบอกได้ว่าคำนั้นสำคัญแค่ไหนจากคำในชุดข้อมูล และกำหนดค่า tfidf เพื่อระบุความสำคัญของคำตามความถี่ TF-IDF Vectorization ทำงานโดยคำนวณ TF และ IDF สำหรับแต่ละคำในเอกสาร จากนั้นจึงใช้ค่าเหล่านี้เพื่อสร้างเวกเตอร์ตัวเลขสำหรับเอกสาร โดยคำนวนจาก term frequency หรือจำนวนครั้งที่แต่ละ word id ปรากฏ ในแต่ละ text หารด้วยจำนวน word ทั้งหมดใน text นั้น (เลยเป็น frequency แทนที่จะเป็น count) แล้วจึงนำมาคูณกับ inverse document frequency หรือจำนวน document ทั้งหมด หารด้วย จำนวน document (หรือ text ที่ผมหมายถึงในบทความนี้) ที่แต่ละ word id ปรากฏอยู่ แล้ว take log เข้าไป (Daroontham, 2561b)

$$TF = \frac{\text{No. of occurence in sentence}}{\text{Total no. of words in sentence}}$$

$$IDF = log\left[\frac{total\ sentences\ in\ document}{sentences\ which\ actually\ _{contain}\ the\ word}\right]$$

$$TF - IDF\ Vectorization = TF\ x\ IDF$$



ภาพประกอบ 1 แสดงกราฟของ *TF-IDF Vectorization*

ที่มา : (Pal et al., 2022)

จากภาพประกอบ 1 อธิบายได้ว่าคำที่มีความสำคัญสูงกว่านั้นจะถูกจัดวางตามลำดับความสำคัญของคำจากน้อยไปมาก และด้วยเหตุนี้จึงเป็นกราฟการเติบโตแบบผกผัน

7. Part of Speech คือการติด tag ว่าคำนั้น ๆ คือคำอะไร เช่น คำนาม, คำกริยา, คำสรรพนาม, คำบุพบท, คำคุณศัพท์ เป็นต้น (Kanoktipsatharporn, 2566)

ศึกษาการทำงานของการประมวลผลภาษาธรรมชาติได้จากตาราง 1

ตาราง 1 แสดงตัวอย่างการทำงานของ NLP

TEXT	LEMMA	POS	TAG	DEP	SHAPE	ALPHA	STOP
Apple	apple	PROPN	NNP	nsubj	Xxxxx	True	False
is	be	AUX	VBZ	aux	xx	True	True
looking	look	VERB	VBG	ROOT	xxxx	True	False
at	at	ADP	IN	prep	xx	True	True
buying	buy	VERB	VBG	pcomp	xxxx	True	False
U.K.	u.k.	PROPN	NNP	compound	X.X.	False	False
startup	startup	NOUN	NN	dobj	xxxx	True	False
for	for	ADP	IN	prep	xxx	True	True
$	$	SYM	$	quantmod	$	False	False
1	1	NUM	CD	compound	d	False	False

ตาราง 1 (ต่อ)

TEXT	LEMMA	POS	TAG	DEP	SHAPE	ALPHA	STOP
billion	billion	NUM	CD	pobj	xxxx	True	False

ที่มา : (spaCy, 2023)

2.3 ทฤษฎีการวิเคราะห์ข้อมูลด้วยหลักการทางคอมพิวเตอร์

2.3.1 การเรียนรู้แบบมีผู้สอน

การสอนให้คอมพิวเตอร์สามารถหาคำตอบได้ด้วยตัวเอง หลังจากเรียนรู้และฝึกหัดจากชุดข้อมูลตัวอย่างไปแล้วระยะหนึ่ง ค่อยมาทดสอบด้วยข้อมูลชุดอื่น เพื่อตรวจสอบว่าแบบจำลองมีความแม่นยำในการทำนายมากน้อยเพียงใด โดยแบ่งออกเป็นการจำแนกประเภทข้อมูล) และ การถดถอย โดยแต่ละปัญหามีการใช้งานเพื่อจุดหมายที่ต่างกัน ซึ่งในงานวิจัยนี้จะใช้เทคนิคการจำแนกประเภทข้อมูล (Achieve.Plus, 2563)

2.3.2 การจำแนกประเภทข้อมูล

เมื่อจำแนกประเภทของข้อมูลตามลักษณะของข้อมูลจะจำแนกเป็น 2 ลักษณะใหญ่ ๆ คือ ข้อมูลเชิงคุณภาพ (Qualitative Data) และข้อมูลเชิงปริมาณ (Quantitative Data)

2.3.2.1 ข้อมูลเชิงคุณภาพ

ข้อมูลที่แสดงถึงสถานภาพ คุณลักษณะ หรือคุณสมบัติ เช่น เพศ เชื้อชาติ สถานภาพสมรส ศาสนา กลุ่มเลือด เป็นต้น (สำนักงานสถิติแห่งชาติ, 2566)

2.3.2.2 ข้อมูลเชิงปริมาณ

ข้อมูลที่อยู่ในรูปตัวเลขที่แสดงถึงปริมาณ อาจเป็นค่าที่ไม่ต่อเนื่อง (Discrete) คือ ค่าที่เป็นจำนวนเต็มหรือจำนวนนับ เช่น จำนวนคนว่างงาน จำนวนพนักงานในบริษัท เป็นต้น หรือเป็นค่าที่ต่อเนื่อง คือค่าที่มีจุดทศนิยมได้ เช่น ความสูง น้ำหนัก อายุ อัตราดอกเบี้ย อัตราเงินเฟ้อ เป็นต้น (สำนักงานสถิติแห่งชาติ, 2566)

สามารถศึกษาความต่างของข้อมูลเชิงคุณภาพและข้อมูลเชิงปริมาณได้จากตาราง 2

ตาราง 2 แสดงความแตกต่างของข้อมูลเชิงคุณภาพและข้อมูลเชิงปริมาณ

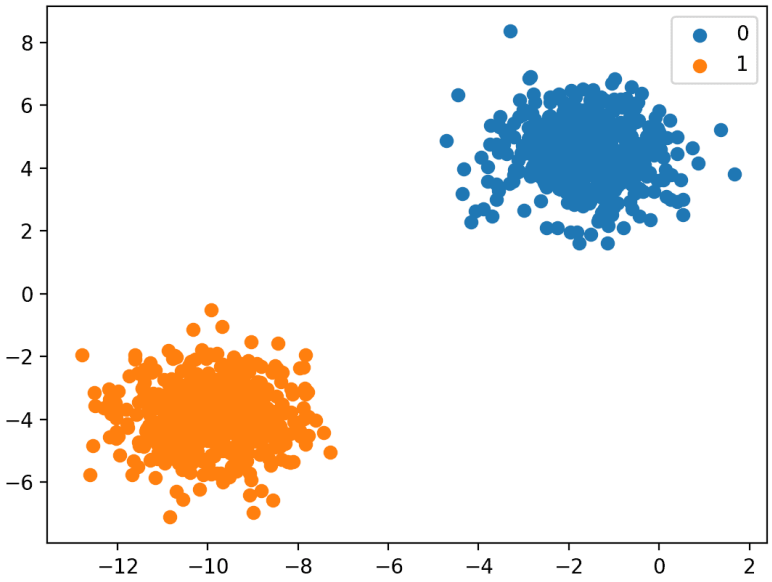
Quantitative data	Qualitative data
ช่วยตอบคำถาม “อะไร” “ที่ไหน” “ อย่างไร” “เมื่อไหร่” และ “ใคร”	ช่วยตอบคำถาม “ทำไม”
ได้ข้อมูลเป็นตัวเลข	ได้ข้อมูลเชิงทัศนคติ ความเห็นและประสบการณ์ที่ผ่าน
ต้องการกลุ่มตัวอย่างจำนวนมาก	ต้องการกลุ่มตัวอย่างจำนวนน้อย
วิเคราะห์ข้อมูลด้วยหลักการทางสถิติ	วิเคราะห์ข้อมูลจากการสัมภาษณ์ และการสังเกต
ใช้คำถามปลายปิด	ใช้คำถามปลายเปิด
ใช้สำหรับการพิสูจนสันนิษฐาน	เพื่อสร้างสมมติฐานใหม่ ๆ หรือสร้างสรรไอเดียใหม่ ๆ

ที่มา : (Kullawattana, 2562)

การจำแนกประเภทข้อมูล (Classification) คือการจำแนกประเภทข้อมูลให้เป็นหมวดหมู่ต่าง ๆ โดยชุดข้อมูลนั้นเป็นชุดข้อมูลที่มีป้ายกำกับ (Labeled Data) ให้กับแบบจำลองการเรียนรู้อัตโนมัติ เพื่อให้แบบจำลองเรียนรู้ความสัมพันธ์ระหว่างข้อมูลและป้ายกำกับ แบบจำลองการเรียนรู้อัตโนมัติจะเรียนรู้จากข้อมูลที่มีป้ายกำกับนี้ เพื่อนำไปใช้ทำนายข้อมูลใหม่ที่ไม่มีป้ายกำกับ เช่น Java developer, DevOps Engineer, Sales เป็นต้น

การสร้างแบบจำลองต้องใช้ชุดข้อมูลการฝึกอบรม (Training set) ที่มีตัวอย่างข้อมูลอินพุตมากพอสมควรเพื่อใช้ในการทำนายผลลัพธ์จากข้อมูลอินพุตนั้น โดยแบ่งออกเป็น 4 ประเภท ดังนี้

- Binary classification (การจำแนกประเภทแบบไบนารี) โดยทั่วไปจะแบ่งประเภทออกเป็น 2 ประเภทคือ “มีงานทำ” หรือ “ว่างงาน” หากเปรียบเป็นตัวเลขก็คือ 0 กับ 1 นั่นเอง ดังภาพประกอบ 2 ซึ่งอัลกอริธึมที่นิยมใช้ ได้แก่
    - Logistic Regression
    - K-Nearest Neighbors
    - Decision Trees
    - Support Vector Machine
    - Naive Bayes



ภาพประกอบ 2 แสดงแผนภูมิกระจายของชุดข้อมูลการจำแนกประเภทไบนารี

ที่มา : (Brownlee, 2020)

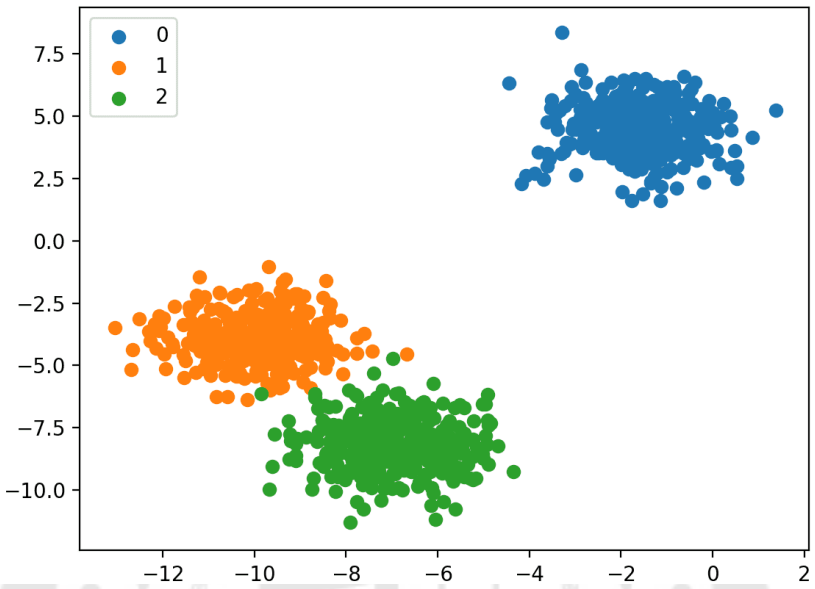
- Multi-Class Classification (การจำแนกประเภทหลายคลาส) ต่างกับการจำแนกประเภทแบบไบนารีตรงที่จะแบ่งประเภทได้มากกว่า 2 ประเภท เช่นการทำนายคำศัพท์หรือประโยคที่อาจจะมีคำหลายหมื่น หรือหลายแสนคำ ดังภาพประกอบ 3 ซึ่งอัลกอริธึมที่นิยมใช้ ได้แก่

- K-Nearest Neighbors
- Decision Trees
- Naive Bayes
- Random Forest
- Gradient Boosting

- อัลกอริธึมการจำแนกประเภทแบบไบนารีสามารถนำมาปรับใช้คู่กับการจำแนกประเภทหลายคลาสได้ โดยอาศัยกลยุทธ์การแปลงคลาสแบบ One-vs-Rest (OvR) หรือ One-vs-One (OvO)

- One-vs-Rest(OvR) การแบ่งปัญหาออกเป็น 1 ต่อ อื่น ๆ ยกตัวอย่าง เช่น หากเรามีหมวดหมู่สีอยู่ 3 หมวดคือ สีแดง สีเขียว และสีฟ้า เราก็จะแบ่งปัญหาการจำแนกแบบไบนารีเป็น 3 ปัญหาคือ 1 จำแนกสีแดงออกจากทั้งหมด 2 จำแนกสีเขียว และ 3 จำแนกสีฟ้า (Achieve.Plus, 2563)

- One-vs-One(OvO) การแบ่งปัญหาออกเป็นคู่ ยกตัวอย่างเช่น หมวดหมู่สีอยู่ 3 หมวดคือ สีแดง สีเขียว และสีฟ้า ก็คือแบ่งปัญหาเป็น 1 จำแนกระหว่าง สีแดงกับสีเขียว 2 จำแนกระหว่าง สีแดงกับสีฟ้า 3 จำแนกระหว่าง สีเขียวกับสีฟ้า (Achieve.Plus, 2563)



ภาพประกอบ 3 แสดงแผนภูมิกระจายของชุดข้อมูลการจำแนกประเภทหลายคลาส

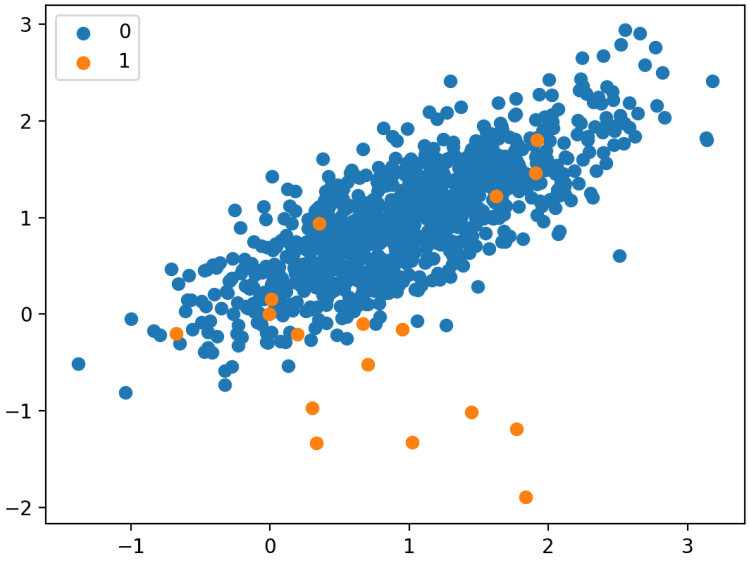
ที่มา : (Brownlee, 2020)

- Multi-Label Classification (การจำแนกประเภทหลายเลเบล) หลายคนอาจจะเข้าใจผิดระหว่างการจำแนกประเภทหลายเลเบลกับการจำแนกประเภทหลายคลาสบ่อย เพื่อเปรียบเทียบให้เข้าใจง่ายขึ้น ขอยกตัวอย่างเช่น รูปภาพรูปหนึ่งสามารถมีรูปดอกไม้ ท้องฟ้า ก้อนเมฆได้ แต่รูปภาพรูปนั้นจะจัดว่าเป็นหมวดหมู่รูปวาด รูปถ่าย หรือรูปเสีย Multi-Label Classification ก็คือการเลเบล หรือติดฉลากว่าในรูปนั้น ๆ มีดอกไม้หรือเปล่า มีก้อนเมฆหรือไม่ ส่วน Multi-Class Classification จะจำแนกว่ารูปนั้นเป็นรูปวาด รูปถ่ายหรือรูปเสีย (Achieve.Plus, 2563)

- Imbalanced Classification (การจำแนกแบบข้อมูลไม่เท่าเทียม) เป็นปัญหาที่เกิดจากการแบ่งชุดข้อมูลในแต่ละคลาสไม่เท่ากัน (Imbalanced dataset) ข้อมูลที่มีค่ามากจะถือว่าเป็นข้อมูลที่ปกติ ส่วนข้อมูลที่มีค่าน้อยเป็นข้อมูลที่ผิดปกติ (Achieve.Plus, 2563) ดังภาพประกอบ 4 มักใช้ในงานด้าน

- Fraud detection

- Outlier detection
- Medical diagnostic tests (Achieve.Plus, 2563) (Brownlee, 2020)



ภาพประกอบ 4 แสดงแผนภูมิกระจายของชุดข้อมูลการจำแนกแบบข้อมูลไม่เท่าเทียม

ที่มา : (Brownlee, 2020)

2.3.3 การทำความเข้าใจกับข้อมูล (Data Understanding)

ผู้วิจัยต้องทำความเข้าใจกับชุดข้อมูลที่นำมาใช้ เช่น ประเภทของข้อมูล จำนวนข้อมูล มีตัวแปรอะไรบ้าง ความหมายของตัวแปรนั้น ๆ เป็นต้น อีกทั้งต้องตรวจสอบความถูกต้องของข้อมูล ความครบถ้วนของข้อมูล ต้องทราบค่าสถิติพื้นฐานของข้อมูล เช่น ค่าเฉลี่ย (Mean), ค่าต่ำสุด (Min), ค่าสูงสุด (Max), ค่าเบี่ยงเบนมาตรฐาน (Standard Deviation) เป็นต้น

2.3.4 การเตรียมข้อมูล (Data Preparation)

เพื่อเตรียมข้อมูลทั้งหมดให้อยู่ในรูปแบบที่สามารถนำไปวิเคราะห์ได้ โดยแบ่งออกเป็น 3 ขั้นตอน ดังนี้

- Data Selection เป็นขั้นตอนการเลือกตัวแปรที่มีความสำคัญ และมีประโยชน์เพื่อนำไปวิเคราะห์ใช้ในงานวิจัย

- Data Cleansing เป็นขั้นตอนตรวจสอบข้อมูลในแต่ละตัวแปรว่ามีข้อมูลที่ผิดปกติ ข้อมูลที่มีค่าว่าง หรือข้อมูลที่มีคำหรือเครื่องหมายสัญลักษณ์พิเศษที่ไม่ถูกต้อง ซึ่งถ้าพบข้อมูลดังกล่าวจะต้องทำการแก้ไข เพิ่มข้อมูล หรือลบข้อมูลให้ถูกต้องครบถ้วน เพื่อนำไปวิเคราะห์ในงานวิจัยต่อไปได้อย่างเหมาะสม

- Data Transformation เป็นขั้นตอนการแปลงข้อมูลของแต่ละตัวแปรให้อยู่ในรูปแบบที่ง่ายต่อการทำงานของเครื่องคอมพิวเตอร์และงานต่อการวิเคราะห์

2.3.5 การสร้างแบบจำลอง (Modeling)

ขั้นตอนการสร้างแบบจำลองการเรียนรู้ของเครื่อง โดยจะนำข้อมูลทั้งหมดมาแบ่งกลุ่มเป็น Training Data หรือ ข้อมูลที่ใช้สำ หรับการเรียนรู้ของแบบจำลอง กับ Testing Data หรือข้อมูลที่ใช้ในการทดสอบกับแบบจำลอง จากนั้นนำมาตรวจสอบความถูกต้องโดยการปรับจูนพารามิเตอร์ของแบบจำลองเพื่อให้ได้ค่าที่ความถูกต้องในระดับที่น่าพอใจ (Anupoomchaiya, 2021)

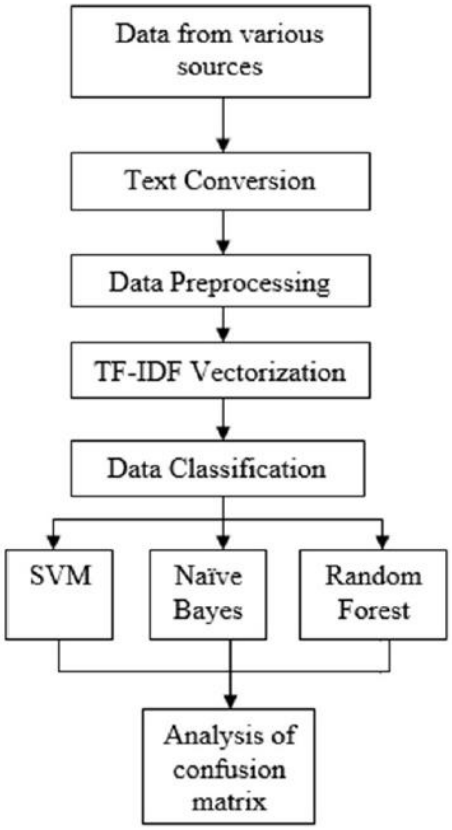
2.3.6 การประเมินผล (Evaluation)

ขั้นตอนประเมินประสิทธิภาพของแบบจำลองการเรียนรู้ของเครื่อง เพื่อค้นหาแบบจำลองที่มีประสิทธิภาพดีที่สุด เพื่อนำแบบจำลองนั้นไปใช้งานต่อไปได้

2.4 งานวิจัยเกี่ยวกับการแนะนำประวัติย่อด้วยหลักการทำงานของเครื่อง

2.4.1 งานวิจัยเรื่อง Resume Classification using various Machine Learning Algorithms (Pal et al., 2022)

งานวิจัยนี้เกิดขึ้นในปี 2022 นำเสนอการจำแนกประเภทของประวัติย่อโดยใช้หลักการเรียนรู้ของเครื่อง (Machine Learning) และประเมินประสิทธิภาพแบบจำลองของ 3 แบบ คือ Nave Bayes, Random Forest, and Support Vector Machine เพื่อช่วยให้กระบวนการสัมภาษณ์ในการคัดเลือกผู้สมัครเป็นไปได้อย่างรวดเร็ว ซึ่งสายงานถูกจัดหมวดหมู่เอาไว้ตามรายละเอียดของตำแหน่งงานแล้ว ในขณะที่สัมภาษณ์งานผู้สัมภาษณ์อาจดูผลการประเมินประสิทธิภาพการคัดกรองประวัติย่อในตำแหน่งงานนั้น ๆ ไปพร้อมกันกับการสัมภาษณ์งาน ซึ่งจะช่วยประหยัดเวลา และลดความผิดพลาดที่เกิดจากมนุษย์ได้อย่างมาก งานวิจัยนี้มีกระบวนการทำงาน 7 ขั้นตอนด้วยกัน ดังภาพประกอบ 5



ภาพประกอบ 5 แผนผังขั้นตอนการทำงาน

ข้อมูลในงานวิจัยนี้เก็บรวบรวมมาจากเว็บไซต์ต่าง ๆ เช่น kaggle.com, glassdoor.com และ indeed.com ชุดข้อมูลนี้เป็นชุดข้อมูลที่ไม่ได้จำแนกประเภทและเป็นชุดข้อมูลที่ไม่มีโครงสร้าง (Unstructured Datasets) ต้องนำข้อมูลมาทำความสะอาดก่อน เมื่อข้อมูลสะอาดแล้วจึงนำข้อมูลเข้าสู่กระบวนการ Preprocessing ด้วยเทคนิคของ NLP เช่น การทำ Tokenization, Stemming, Lemmatization, POS Tagging และ TF-IDF Vectorization เป็นต้น จะได้ข้อมูลที่มี 10,000 แถว โดยแบ่งออกมาเป็น 3 คอลัมน์คือ Query (ประเภทงาน), Description (ข้อมูลดิบในประวัติย่อ เช่น การศึกษา ทักษะและประสบการณ์การทำงาน) และ text_final (ข้อมูลที่เข้าสู่กระบวนการ NLP เรียบร้อยแล้ว) จากนั้นเข้ากระบวนการสร้างแบบจำลอง โดยแบ่งข้อมูลออกเป็น 70% สำหรับข้อมูลการฝึกอบรม (Training Data) และ 30% สำหรับข้อมูลการทดสอบ (Test Data)

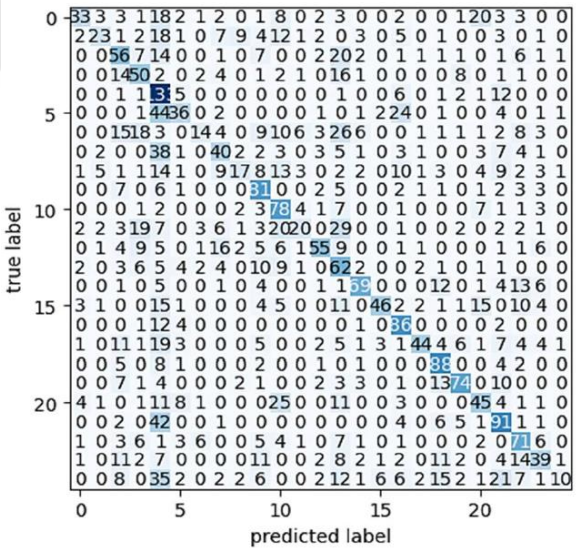
สร้างแบบจำลองการจำแนกประเภท 3 แบบ เพื่อประเมินประสิทธิภาพของการทดลอง ได้แก่ Naïve bayes classification ได้ Accuracy 45% ในขณะที่ Support Vector

Machine (SVM) ได้ Accuracy 60% ซึ่งดีกว่า Naïve bayes และ Random Forest ได้ Accuracy 70% จากตาราง 3 แสดงให้เห็นว่า Random Forest นอกจากจะมีค่า Accuracy สูงที่สุดแล้ว ยังมีค่า Precision, Recall, F1 score อยู่ที่ 0.687, 0.683 และ 0.678 ตามลำดับ

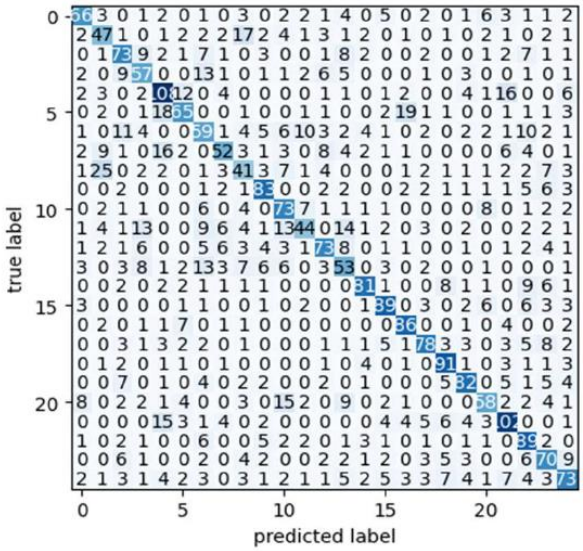
ตาราง 3 แสดงตารางเปรียบเทียบ Confusion Matrix ของ NB Classifier, SVM, RF

Algorithm	Accuracy	Precision	Recall	F1 Score
Naïve Bayes	45	0.521	0.452	0.448
SVM	60	0.598	0.597	0.594
Random Forest	70	0.687	0.683	0.678

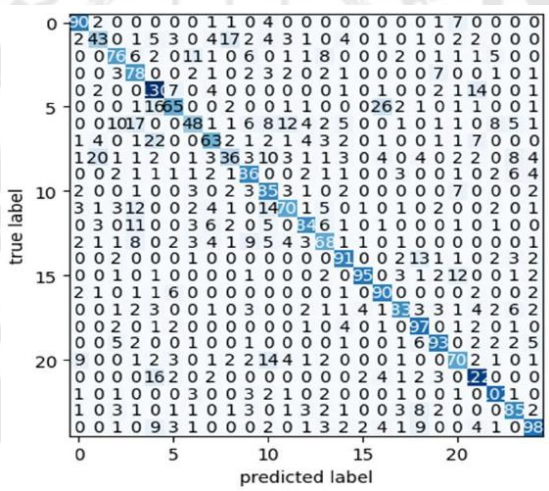
ผลการทดลองของงานวิจัยนี้แสดงให้เห็นว่า Random Forest นอกจากจะมีค่า Accuracy สูงที่สุดแล้ว ยังมีค่า Precision, Recall, F1 score สูงด้วยเช่นกัน อยู่ที่ 0.687, 0.683 และ 0.678 ตามลำดับ ดังภาพประกอบ 10 ซึ่งความแม่นยำของ Random Forest สามารถปรับปรุงได้โดยการเพิ่มจำนวน Decision Tree หรือ โดยการเพิ่มขนาดของข้อมูล ในงานวิจัยนี้ใช้ 2,500 ต้น ถึงแม้ว่าแบบจำลองนี้จะมีประสิทธิภาพที่ดี แต่ในขณะเดียวกันก็ใช้เวลาในการเรียนรู้นานกว่า และอีกเทคนิคหนึ่งที่สามารถนำไปใช้เพื่อปรับปรุงประสิทธิภาพได้คือ การใช้การปรับแต่งไฮเปอร์พารามิเตอร์สำหรับ Random Forest Classifier



ภาพประกอบ 6 *Confusion Matrix for Naïve Bayes Classification*



ภาพประกอบ 7*Confusion Matrix for SVM Classification*



ภาพประกอบ 8 *Confusion Matrix for Random Forest Classification*

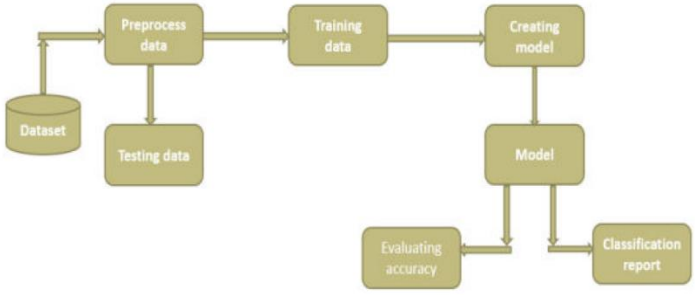
2.4.2 งานวิจัยเรื่อง Resume Screening Using Machine Learning

งานวิจัยนี้เกิดขึ้นในปี 2022 นำเสนอวิธีการคัดกรองประวัติย่อด้วยเทคนิคการประมวลผล

ภาษาธรรมชาติ (NLP) โดยใช้ไลบรารี Python ที่ชื่อ Natural Language Toolkit (NLTK) และประเมินประสิทธิภาพของแบบจำลองเพื่อหาแบบจำลองที่เหมาะสมที่สุด เพื่อช่วยเพิ่มประสิทธิภาพในการสรรหาบุคลากรสายไอที โดยงานวิจัยนี้เปรียบเทียบประสิทธิภาพของแบบจำลองดังนี้ K-Nearest Neighbor (KNN), Support Vector Machine (SVM), Logistic

Regression (LR), Multi-Layer Perceptron (MLP) ซึ่ง เป็นประเภทที่ง่ายที่สุดของโครงข่ายประสาทเทียมที่มีโครงสร้างเป็นแบบหลาย ๆ ชั้น ใช้สำหรับงานที่มีความซับซ้อนได้ผลเป็นอย่างดี โดยมีกระบวนการฝึกฝนเป็นแบบมีผู้สอน และใช้ขั้นตอนการส่งค่าย้อนกลับ (Backpropagation) สำหรับการฝึกฝน (วิกิพีเดีย, 2566)

แผนผังสถาปัตยกรรมของงานวิจัยนี้ ดังภาพประกอบ 9 ผู้สมัครอัปโหลดประวัติย่อของตนเข้ามา จากนั้นประวัติย่อจะถูกส่งไปยังตัวแยกวิเคราะห์ (resume parser) ซึ่งเป็นไปป์ไลน์ของอัลกอริธึม NLP ที่แยกข้อมูลที่สำคัญออกมาจากประวัติย่อ จากนั้นทำการเพิ่มมูลค่า (add value) ให้กับข้อมูลที่ดึงออกมาจากข้อมูลเวกเตอร์ที่เราแปลงคำให้เป็นเวกเตอร์แล้ว และสุดท้ายส่งไปยังแบบจำลองการเรียนรู้แต่ละแบบเพื่อประเมินประสิทธิภาพที่เหมาะสมกับปัญหาการคัดกรองประวัติย่อ

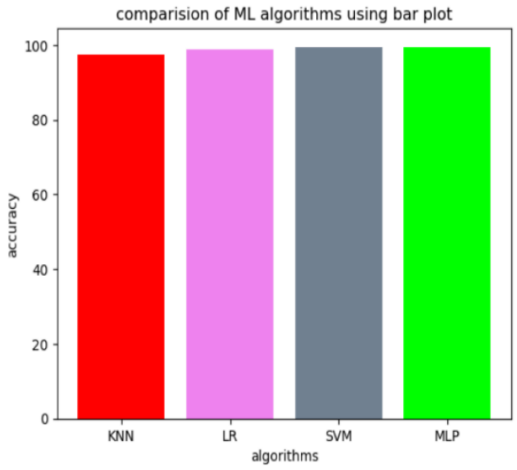


ภาพประกอบ 9 *Architecture Diagram*

สรุปการประเมินผลของแบบจำลอง KNN, SVM, LR และ MLP พบว่าค่า Accuracy ของ SVM และ MLP อยู่ที่ 99.48% มากกว่า KNN และ LR ดังตาราง 4 และภาพประกอบ 10

ตาราง 4 เปรียบเทียบค่า Accuracy ของแบบจำลอง KNN, LR, SVM, MLP

	model	accuracy
0	KNN_acc	97.4093
1	LR_acc	98.9637
2	SVM_acc	99.4819
3	MLP_acc	99.4819



ภาพประกอบ 10 *Comparison of ML Algorithms Using Bar Plot*

2.4.3 งานวิจัยเรื่อง Resume recommendation based on text similarity (Dong, 2023)

งานวิจัยนี้เกิดขึ้นในปี 2022 นำเสนอวิธีการแนะนำประวัติย่อโดยมีความพยายามที่จะจับคู่ประวัติย่อของผู้หางานกับตำแหน่งงานแบบอัตโนมัติโดยพิจารณาจากทักษะและข้อมูลต่าง ๆ ในประวัติย่อนำมาวิเคราะห์และแนะนำว่าเหมาะสมกับตำแหน่งงานอะไรโดยอิงตามการจำแนกข้อความ โดยเฉพาะอย่างยิ่งการเข้ารหัสและดึงข้อมูลคุณลักษณะของประวัติย่อที่แตกต่างกัน และการให้คะแนนความตรงกันระหว่างข้อมูลผู้หางานกับข้อมูลงานโดยอิงตามอัลกอริธึม XGBoost ซึ่งชุดข้อมูลถูกเก็บรวบรวมจากอินเทอร์เน็ตเป็นประวัติย่อจริงและไม่ซ้ำกัน 30,000 ฉบับ นำมาทำความสะอาดและเข้าสู่กระบวนการ preprocessing โดยใช้เทคนิค NLP เพื่อสกัดคำต่าง ๆ ออกมาก่อน จากนั้นเข้าสู่แบบจำลอง XGBoost หลังจากการฝึกแบบจำลองก็ได้ผลลัพธ์การฝึกอบรม 53 รายการ และเลือก 5 รายการที่มีข้อมูลจำนวนมาก

ตาราง 5 Predicted results for most common jobs.

Label	Precision	Recall	F1-score	Support
Database Administrator	0.72293	0.80496	0.76174	282
Front End Developer	0.66000	0.78261	0.71609	253
IT Security Analyst	0.53629	0.61290	0.57204	217
Network Administrative	0.53182	0.49576	0.51316	236
System Administrative	0.44510	0.51546	0.47771	291

จากตาราง 5 และ 6 แสดงให้เห็นว่าความแม่นยำ (Precision) ของแบบจำลองสำหรับ Database Administrator คือ 0.72293, Recall คือ 0.80496 และ f1-score คือ 0.76174 สูงที่สุดเมื่อเทียบกับอีก 5 รายการ ซึ่งสูงกว่าผลการทำนายของ IT Security Analyst ประมาณ 20% และยังแสดงให้เห็นว่าประวัติย่อของผู้สมัครงานตำแหน่ง Database Administrator และ Front End Developer นั้นมีความเป็นเอกลักษณ์เฉพาะ และสามารถระบุได้จากข้อความประวัติย่อของพวกเขาได้อย่างง่ายดาย

ตาราง 6 Average predicted results.

	Precision	Recall	F1-score	Support
Accuracy			0.57891	1432

ตาราง 6 (ต่อ)

	Precision	Recall	F1-score	Support
Macro avg	0.07330	0.07342	0.07167	1432
Weighted avg	0.52652	0.57891	0.54883	1432

บางผลลัพธ์แสดงให้เห็นว่าอัตราความแม่นยำนั้นต่ำเนื่องจากปริมาณข้อมูลค่อนข้างน้อย เช่น ตำแหน่งของ Network Engineer มีข้อมูลอยู่ 25 รายการ ทำนายได้ถูกต้องเพียง 42.85% ซึ่งอัตราความแม่นยำจะสูงเมื่อมีข้อมูลจำนวนมาก อีกทั้งยังพบว่าผลลัพธ์บางอย่างมีความลำเอียงเนื่องจากข้อมูลประวัติย่อแบบเดียวกันแต่มีหลาย label และข้อมูลบางส่วนมีข้อมูลจำนวนมาก แต่ความแม่นยำไม่สูงนัก เนื่องจากข้อมูลของสอง label มีความสอดคล้องกันอย่างมาก

2.4.4 งานวิจัยเรื่อง Machine Learned Job Recommendation (Paparrizos et al., 2011)

งานวิจัยนี้เกิดขึ้นในปี 2011 นำเสนอการแก้ไขปัญหาการแนะนำงานที่เหมาะสมให้กับผู้ที่กำลังมองหางานใหม่ ผู้วิจัยจัดรูปแบบปัญหาการแนะนำนี้เป็นปัญหาการเรียนรู้แบบ supervised machine learning โดยนำข้อมูลการเปลี่ยนงานในอดีตทั้งหมด รวมถึงข้อมูลที่เกี่ยวข้องกับพนักงานและองค์กรเพื่อคาดการณ์การเปลี่ยนงานครั้งต่อไปของพนักงาน ผู้วิจัยฝึกแบบจำลองการเรียนรู้ของเครื่องที่สกัดจากโปรไฟล์พนักงานที่เผยแพร่ต่อสาธารณะบนเว็บไซต์ ซึ่ง

ประกอบไปด้วย 3 คอลัมน์ได้แก่ 1. ข้อมูลส่วนตัวของผู้หางาน เช่น ชื่อ นามสกุล ข้อมูลทางภูมิศาสตร์ 2. ประสบการณ์การทำงาน เช่น ชื่อองค์กร ชื่อตำแหน่งงาน วันที่เริ่มงาน ขนาดขององค์กร ประเภทธุรกิจขององค์กร เป็นต้น 3. ข้อมูลการศึกษาของผู้หางาน เช่น ชื่อมหาวิทยาลัย ระดับการศึกษา คณะที่เรียน วันที่เริ่มต้นและสิ้นสุด เป็นต้น ข้อมูลทั้งหมดที่ใช้ในงานวิจัยนี้จำนวน 5.29 ล้านรายการ จำนวนองค์กรที่ไม่ซ้ำกัน 1.27 ล้านรายการ จำนวนมหาวิทยาลัยที่ผู้หางานจบการศึกษาที่ไม่ซ้ำกัน 1.95 แสนรายการ ซึ่งจำนวนที่ไม่ซ้ำกันนี้ได้มาจากการทำความสะอาดข้อมูลและไม่สนใจ (ignore) ข้อมูลที่พบครั้งเดียว ดังตาราง 7

ตาราง 7 ฟีเจอร์ที่ใช้ในแบบจำลองการทำนาย

Feature type	Feature	Range
Institution	company title	String
	industry	String
	company type	{public, private}
	number of employees	Z
employee	number of jobs	Z
	position title	String
	best position title	String
	years of experience	Z
	number of universities	Z
	educational degree	String

โดยแบ่งข้อมูลออกเป็น 3 ชนิด ดังตาราง 8

- Setup I มหาวิทยาลัยแรก + 100 องค์กรแรก โดยที่จำนวน Train set 70%, Test set 30%
- Setup II 100 องค์กรแรก โดยที่จำนวน Train set 70%, Test set 30%
- Setup III 25 องค์กรแรก โดยที่จำนวน Train set 70%, Test set 30%

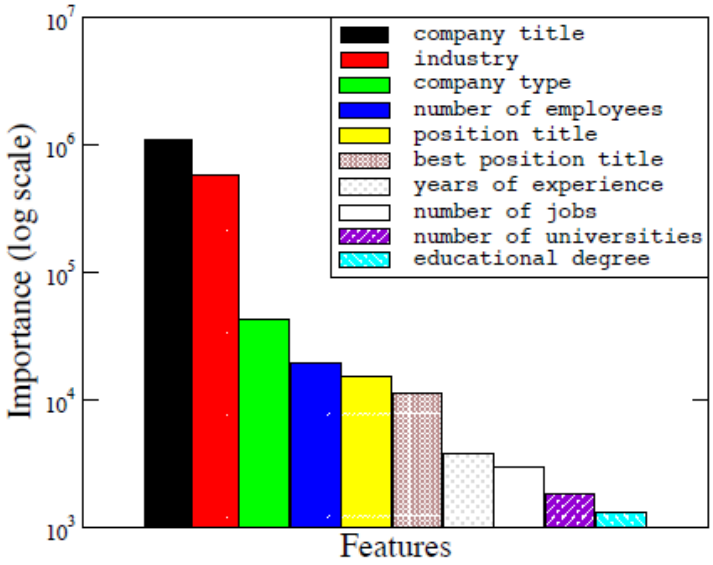
ตาราง 8 การตั้งค่าที่ใช้ในการทดลอง

Setup	Data sample	Train set sizes	Test set sizes
I	Top 100 universities + Top 100 companies	65,622	28,124
II	Top 100 companies	52,142	22,346
III	Top 25 companies	45,891	19,668

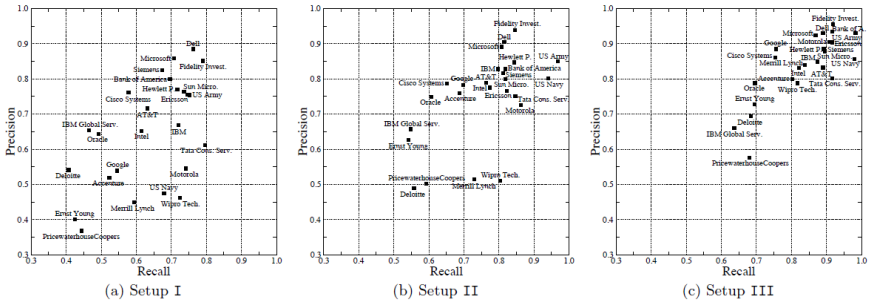
ในการฝึกและประเมินแบบจำลองการเรียนรู้ของเครื่อง ผู้วิจัยใช้ชุดเครื่องมือ Weka machine-learning โดยทดลองใช้หลายอัลกอริธึมการเรียนรู้ของเครื่อง แต่นำเสนอผลลัพธ์เฉพาะสำหรับตัวจัดประเภทไฮบริด decision table/naive Bayes (DTNB)

ตาราง 9 แสดงความถูกต้อง (Accuracy) ในการทำนาย

Setup	Acc_Baseline (%)	Acc_DTNB (%)	Difference
I	15.21	66.78	51.57
II	15.40	78.26	62.86
III	15.97	86.09	70.12



ภาพประกอบ 11 แสดง *Feature importance* ของแต่ละข้อมูล

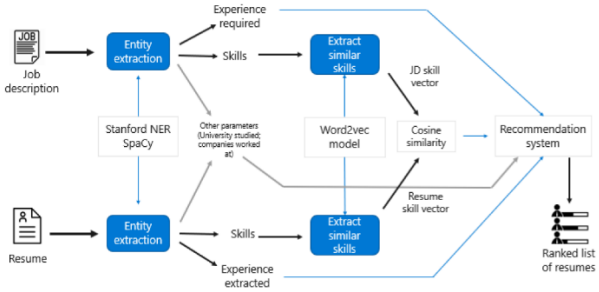


ภาพประกอบ 12 ค่า *Precision and recall* ของข้อมูลทั้ง 3 แบบ

ผลการทำลองพบว่าค่าความถูกต้อง (Accuracy) ของข้อมูลแบบที่ III มีค่าสูงสุด เมื่อเทียบกับทั้ง 3 แบบ ค่า Accuracy อยู่ที่ 66.8%, 78.2% และ 86.0% สำหรับการตั้งค่า I, II และ III ตามลำดับ โดยตั้งค่า baseline ที่ 15% ดังตาราง 9 ส่วนค่า Feature importance ซึ่งบ่งบอกถึงคุณลักษณะที่มีความสำคัญต่อการคาดการณ์เรียงลำดับมากที่สุดไปน้อยที่สุด พบว่า ชื่อบริษัทและอุตสาหกรรมเป็นคุณลักษณะที่สำคัญต่อการทำนายผล ดังภาพประกอบ 11 และจากภาพประกอบ 12 ยิ่งทำให้เห็นว่าค่า Precision และ Recall ของข้อมูลแบบ III มีค่าสูงที่สุดเช่นเดียวกัน

2.4.5 งานวิจัยเรื่อง Differential Hiring using a Combination of NER and Word Embedding (Suhas & Manjunath, 2020)

งานวิจัยนี้เกิดขึ้นในปี 2020 นำเสนอวิธีการจับคู่ผู้หางานที่มีความสามารถที่เหมาะสมที่สุดกับงาน ด้วยหลักการของ NER, Word embedding model ชนิด word2vec และ Cosine similarity ซึ่งข้อมูลนี้ได้มาจากการรวบรวมเอกสารที่เกี่ยวข้องกับปัญหาในการหางาน รวมถึงชุดประวัติย่อของผู้สมัครที่ผ่านการอนุมัติแล้ว สามารถดูแผนผังการทำงานทั้งหมดในงานวิจัยนี้ ดังภาพประกอบ 13



ภาพประกอบ 13 แผนผังการทำงานของงานวิจัยนี้

กระบวนการทำงานในงานวิจัยนี้แบ่งออกเป็น 3 ส่วนหลักคือ 1. NER จะทำหน้าที่ในการหาตำแหน่ง และจัดหมวดหมู่ของกลุ่มคำ ที่อยู่ในเอกสาร 2. จากนั้นทำการแปลงคำให้เป็นเวกเตอร์ตัวเลข และสร้างแบบจำลอง word2vec เพื่อเรียนรู้ความสัมพันธ์ระหว่างคำหรือข้อความในเอกสารที่ได้จากการคำนวนเวกเตอร์ตัวเลข และ 3. ใช้ Cosine Similarity หาความคล้ายคลึงกันของคำหรือข้อความ ผู้วิจัยแบ่งการทดสอบประสิทธิภาพออกเป็น 4 ชนิดได้แก่

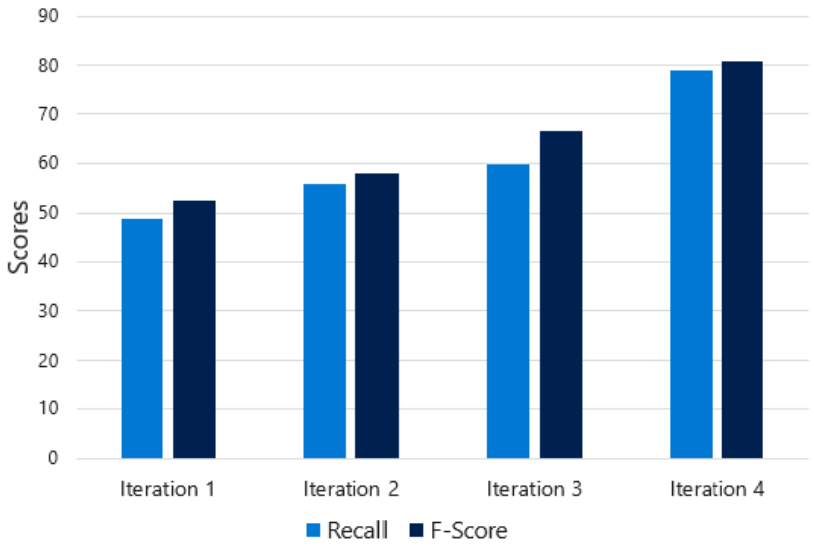
Iteration 1: นำข้อมูลประวัติย่อ 100 ฉบับที่ทำการ preprocessing เล็กน้อย เช่น ลบเครื่องหมายวรรคตอน และนำไปฝึกอบรมในแบบจำลอง NER

Iteration 2: นำข้อมูลไปทำ Text preprocessing

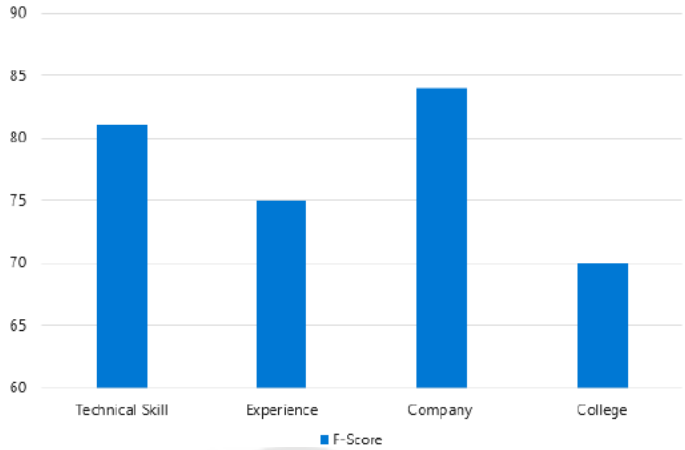
Iteration 3: เพิ่มชุดข้อมูลจาก 100 เป็น 200 เพื่อนำไปฝึกอบรม

Iteration 4: ปรับปรุงประสิทธิภาพหลายอย่าง เช่น เพิ่ม window size, เพิ่มคำระหว่างการฝึกอบรม NER

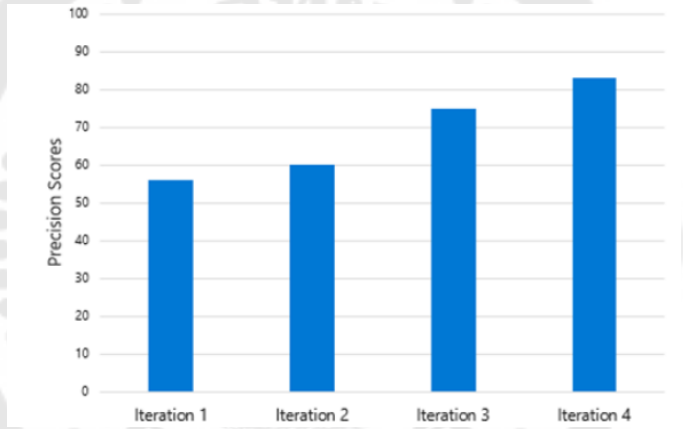
จากผลการทดลองแสดงให้เห็นว่าชุดข้อมูล Iteration 4 มีค่า Precision, Recall, F-Score สูงที่สุด ส่วนคำว่า Experience มีค่า Precision ต่ำที่สุด เพราะข้อมูลประสบการณ์การทำงานมีความแตกต่างกันในแต่ละประวัติย่ออย่างมากจึงส่งผลให้แบบจำลองไม่สามารถระบุได้อย่างแม่นยำ ดังภาพประกอบ 14-16



ภาพประกอบ 14 เปรียบเทียบค่า Recall และ F-Score



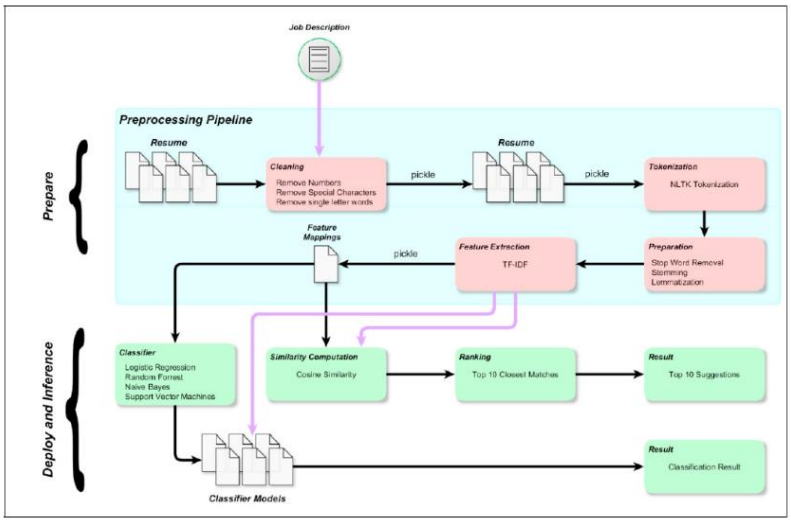
ภาพประกอบ 15 เปรียบเทียบค่า *F-Score*



ภาพประกอบ 16 เปรียบเทียบค่า Precision

2.4.6 งานวิจัยเรื่อง A Machine Learning approach for automation of Resume Recommendation system (Roy et al., 2020)

งานวิจัยนี้เกิดขึ้นในปี 2020 จุดประสงค์เพื่อค้นหาประวัติย่อของผู้หางานจากคลังข้อมูลขององค์กรเพื่อพบผู้หางานที่เหมาะสมกับงาน โดยใช้ชุดข้อมูลสาธารณะที่หามาจากในเว็บไซต์ kaggle มีทั้งหมด 3 คอลัมน์ ได้แก่ ID, Category (ประเภทงาน) และ Resume (ประวัติย่อของผู้หางาน) โดยแบ่งประเภทงานออกเป็น 24 ประเภท ผู้วิจัยได้พัฒนาโซลูชันที่ใช้การเรียนรู้ของเครื่อง โดยเปรียบเทียบความถูกต้องของแบบจำลองทั้ง 4 ได้แก่ Random Forest (RF), Multinomial Naive Bayes (NB), Logistic Regression (LR) และ Linear Support Vector Classifier (Linear SVC) สามารถศึกษาโครงสร้างการทำงานของแบบจำลอง ดังภาพประกอบ 17

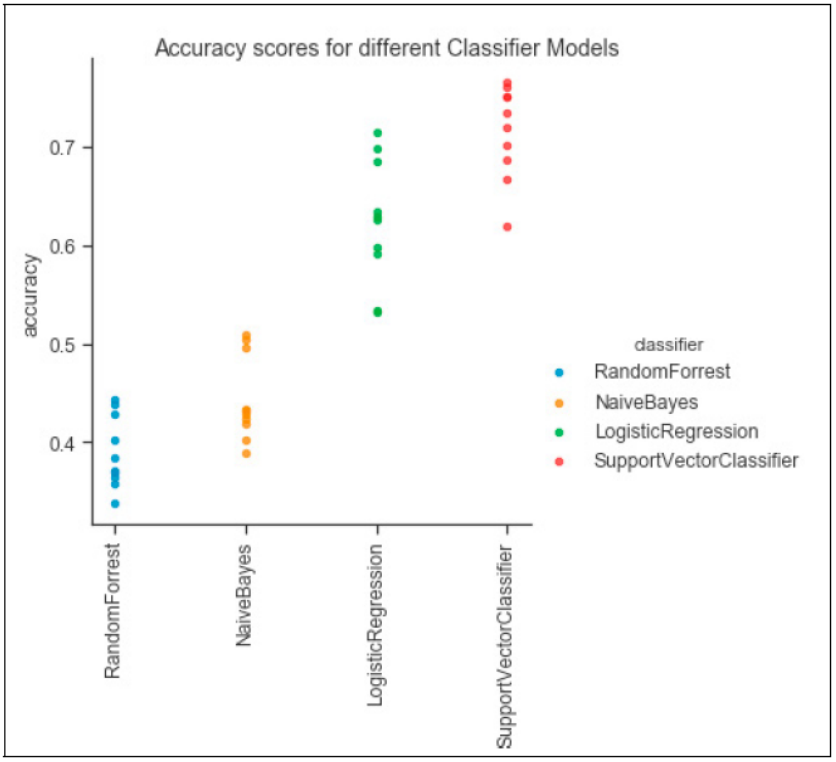


ภาพประกอบ 17 โครงสร้างการทำงานของแบบจำลองที่ใช้ในงานวิจัย

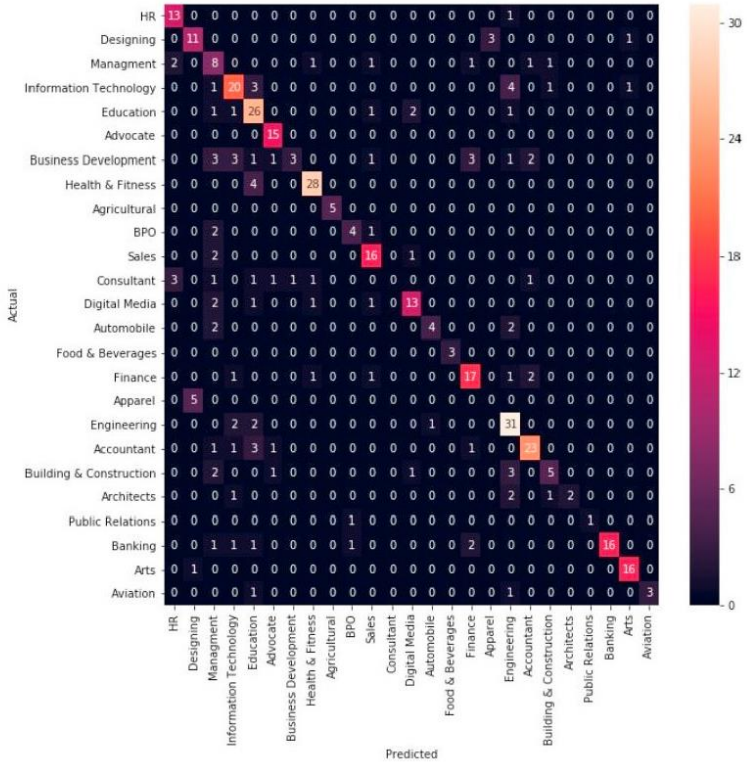
ผลการทดลองพบว่า Random Forest classifier มีค่า Accuracy น้อยที่สุด อยู่ที่ 38.99% และ Linear Support Vector Classifier (SVC) มีค่า Accuracy ที่ได้จากการหาค่า Cross Validation จำนวน 10 fold สูงที่สุด อยู่ที่ 78.53% ศึกษาผลลัพธ์ได้จาก ตาราง 10 และ ภาพประกอบ 18, 19

ตาราง 10 ค่า Accuracy ของแบบจำลองต่าง ๆ

Classification	Accuracy
Random Forest	0.3899
Multinomial Naïve Bayes	0.4439
Logistic Regression	0.6240
Linear Support Vector Machine Classifier	**0.7853**



ภาพประกอบ 18 ค่า Accuracy ของแบบจำลองต่าง ๆ



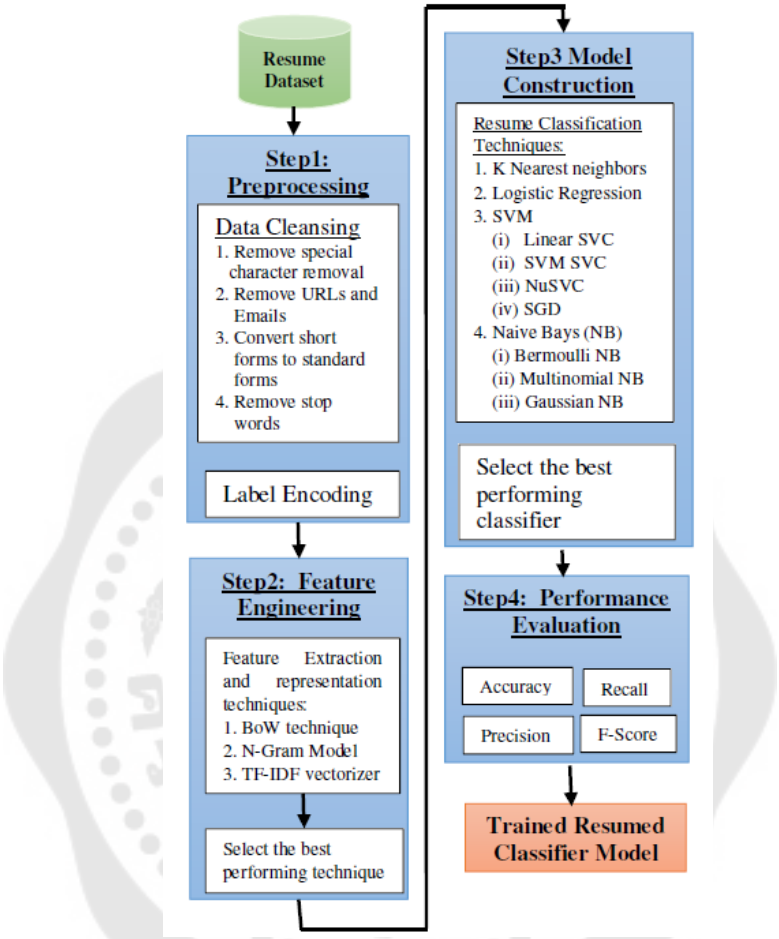
ภาพประกอบ 19 ค่า *Confusion Matrix* ของแบบจำลอง *Linear SVC*

2.4.7 งานวิจัยเรื่อง A Machine Learning approach for automation of Resume Recommendation system (Ali et al., 2022)

งานวิจัยนี้เกิดขึ้นในปี 2022 นำเสนอระบบการจำแนกประเภทของประวัติย่อด้วยการเรียนรู้ของเครื่องทั้ง 9 แบบจำลอง ดังนี้ ดังภาพประกอบ 20

- K-Nearest Neighbor
- Logistic Regression
- Support Vector Machine
    - Linear SVC
    - SVM, SVC
    - NuSVC
    - SGD
- Naïve Bayes
    - Bernoulli NB

- Multinomial NB
- Gaussian NB

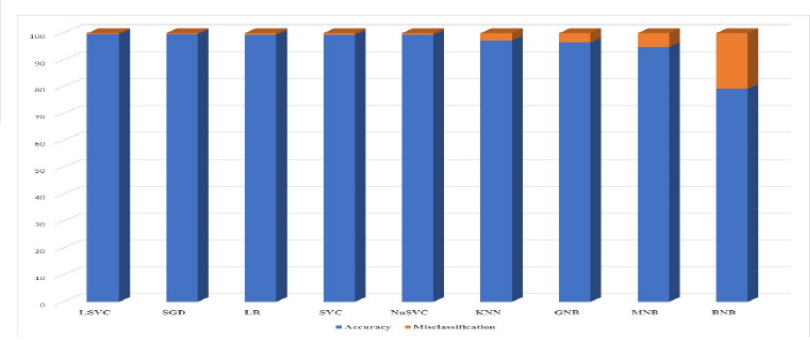


ภาพประกอบ 20 วิธีการจำแนกประเภทประวัติย่อในงานวิจัย

ชุดข้อมูลที่ใช้ในงานวิจัยนี้มีทั้งหมด 962 รายการที่ถูก label แยกตามประเภทงานได้ 25 ประเภท และข้อมูลของประวัติย่อที่ต้องนำมาทำความสะอาด และเข้าสู่กระบวนการ Preprocessing โดยใช้การประมวลผลภาษาธรรมชาติ (NLP) เพื่อจัดการกับคำและประโยคที่อยู่ในประวัติย่อ จากนั้นจึงเข้าสู่แบบจำลองต่าง ๆ เพื่อประเมินผลการทดลอง โดยแบ่งข้อมูล 70% Train set, 30% Test set และใช้ตัวชี้วัดประสิทธิภาพคือ Overall accuracy, Precision, Recall, F-Score matrices ได้ผลลัพธ์ ดังตาราง 11

ตาราง 11 การประเมินประสิทธิภาพของแบบจำลองการเรียนรู้ของเครื่อง

Classification	Precision	Recall	F-Score	Overall Accuracy %	Misclassification %
LSVC	1.00	1.00	1.00	99.6	0.4
SGD	1.00	1.00	1.00	99.6	0.4
LR	1.00	0.99	0.99	99.3	0.7
SVC	1.00	0.99	0.99	99.3	0.7
NuSVC	0.99	0.99	0.99	99.3	0.7
KNN	0.99	0.98	0.99	97.2	2.8
GNB	0.98	0.96	0.96	96.5	3.5
MNB	0.98	0.95	0.96	94.8	5.2
BNB	0.89	0.76	0.79	79.2	20.8



ภาพประกอบ 21 แสดงค่า Overall Accuracy กับ Misclassification

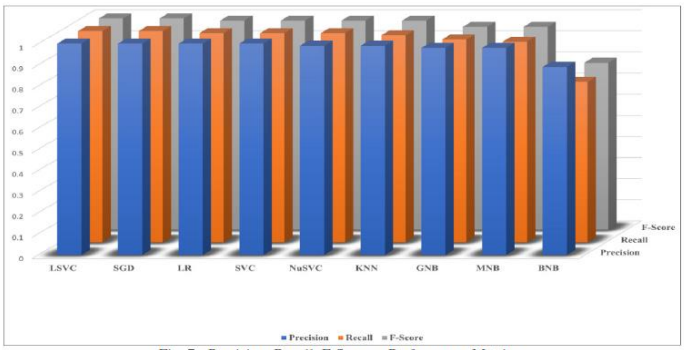
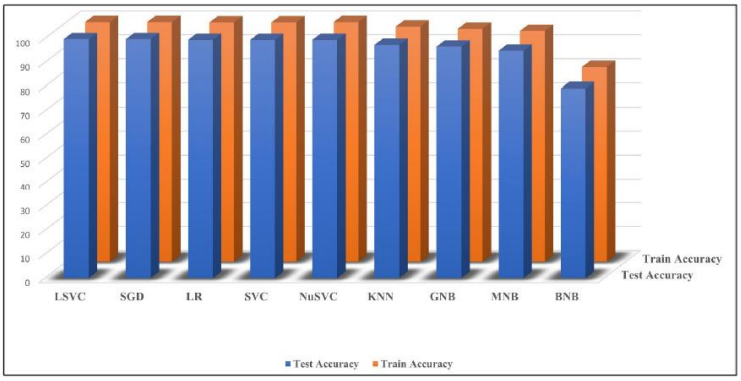


Fig. 7: Precision, Recall, F-Score, - Performance Matrices

ภาพประกอบ 22 แสดงค่า Precision, Recall, F-Score

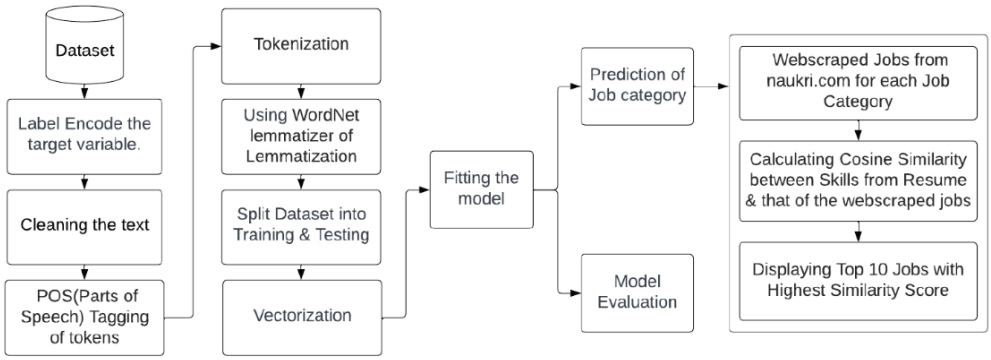


ภาพประกอบ 23 ค่า Train vs Test Accuracy

จากภาพประกอบ 21-23 พบว่าแบบจำลองตระกูล Support Vector Machine (Linear SVC, SGD, LR, SVC, NuSVC) มีค่า Accuracy, Precision, Recall, F-Score สูงที่สุด ส่วน Bernoulli NB มีค่า Accuracy, Precision, Recall, F-Score ต่ำที่สุด

2.4.8 งานวิจัยเรื่อง Resume Analysis and Job Recommendation (Mankawade et al., 2023)

งานวิจัยนี้เกิดขึ้นในปี 2023 นำเสนอระบบการแนะนำงาน เพื่อเพิ่มประสิทธิภาพในการค้นหาตำแหน่งงานของผู้หางาน ผู้วิจัยได้เสนอสถาปัตยกรรมเพื่อจำแนกประเภทงานที่เหมาะสมที่สุดตามประวัติย่อของผู้หางานคนนั้น ๆ และใช้การประมวลผลภาษาธรรมชาติ (NLP) และการเรียนรู้ของเครื่องเพื่อฝึกอบรมแบบจำลองที่สามารถทำนายตำแหน่งงาน โดยนำข้อมูลตำแหน่งงานมาจากเว็บไซต์ Naukri.com ทักษะที่จำเป็นของงานจะถูกจับคู่กับทักษะของแต่ละบุคคลโดยใช้อัลกอริธึม cosine similarity จัดอันดับและแสดงตำแหน่งงานที่แนะนำให้กับผู้หางาน



ภาพประกอบ 24 แผนผังการทำงานของงานวิจัยนี้

ชุดข้อมูลที่ใช้ในงานวิจัยนี้มาจาก Kaggle ชื่อ "Resume dataset" ประกอบด้วย 962 ประวัติย่อ โดยมีคอลัมน์ Resume รวมข้อมูลต่าง ๆ ของผู้สมัครงานไว้ และคอลัมน์ Category มีการจัดประเภทหมวดหมู่งานที่แตกต่างกันไว้แล้วจำนวน 25 หมวดหมู่ จากนั้นนำชุดข้อมูลนี้ไปเข้ากระบวนการ Data Preprocessing โดยใช้หลักการทำงานของ NLP เช่น การทำ Tokenization, Lemmatization, POS tagging เป็นต้น ก่อนจะนำไปเข้าฝึกอบรมในแบบจำลอง multinomial NB และใช้ตัวชี้วัดประสิทธิภาพเพื่อดูค่า Accuracy, Precision, Recall และ f1-score ในส่วนของระบบการแนะนำงานนั้น ใช้วิธีไปดึงข้อมูลตำแหน่งงานจากเว็บไซ์ naukri.com และใช้หลักการของ cosine similarity เพื่อเปรียบเทียบความคล้ายคลึงกันของทักษะในตำแหน่งงานกับทักษะของผู้สมัครงาน จากนั้นแนะนำ 15 ตำแหน่งงานที่เหมาะสมกับผู้สมัครมากที่สุด ดังภาพประกอบ 24

จากผลการทดลองแสดงให้เห็นว่าค่า Accuracy, Precision, Recall ค่า Accuracy ของอัลกอริธึม Naïve Bayes 97%, macro f1 score is 96% และสามารถใช้หลักการของ cosine similarity เพื่อแนะนำตำแหน่งงานที่ตรงกับผู้สมัครคนนั้น ๆ ได้ ดังตาราง 12, 14 อีกทั้งยังทดลองนำชุดข้อมูลของ Email Spam มาวัดประสิทธิภาพ ดังตาราง 13

ตาราง 12 Classification Report for Resume Dataset

	Precision	Recall	F1-Score	Support
accuracy			0.97	241
macro avg	0.99	0.95	0.96	241
Weighted avg	0.97	0.97	0.96	241

ตาราง 13 Classification Report for Email Dataset

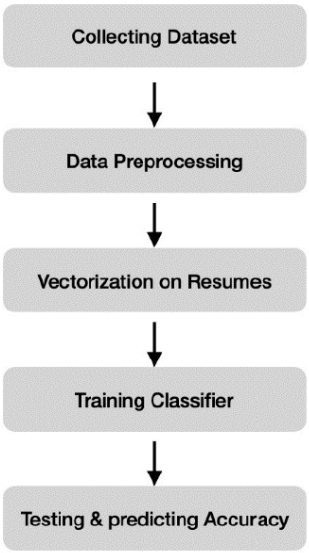
	Precision	Recall	F1-Score	Support
accuracy			0.98	1393
macro avg	0.97	0.96	0.96	1393
Weighted avg	0.98	0.98	0.98	1393

ตาราง 14 แนะนำตำแหน่งงาน

Sr.No.	Title	Company	Skills	Professions
1.	Senior Business Analyst	Freshworks	[python],[analytical],[Business] …	Data Science
2.	Analyst-Data Science	Accenture	[consulting],[MySql],[Data Science]…	Data Science
3.	Analyst, Data Science & Analyst	TransUnion	[SQL],[Communication],[IT Skills]…	Data Science
4.	Data Science Manager	Rapido	[Data Science],[Data Analysis],[NoSQL]…	Data Science

2.4.9 งานวิจัยเรื่อง Resume Classification using Elite Bag-of-Words Approach (Sharma et al., 2023)

งานวิจัยนี้เกิดขึ้นในปี 2023 นำเสนอการคัดกรองผู้สมัครงานด้วยเทคนิคการประมวลผลภาษาธรรมชาติ (NLP) และเทคนิค text vectorization ที่เรียกว่า Elite bag-of-words สำหรับข้อมูลในประวัติย่อ เพื่อนำคำแต่ละคำไปจัดอันดับ หาความถี่ของคำและจัดอันดับในแต่ละคลาสด้วยวิธี maximum entropy partitioning (MEP) และทำนายประเภทของประวัติย่อตามสายงานที่เกี่ยวข้อง ผู้วิจัยเลือกใช้การประเมินประสิทธิภาพโดยการเปรียบเทียบค่า Accuracy ของ 4 วิธี ได้แก่ One-hot encoding, TF-IDF, TF และ Elite keywords สามารถดูแผนผังการทำงานจากภาพประกอบ 25



ภาพประกอบ 25 แผนผังการทำงานของงานวิจัย

หลังจากนำข้อมูลเข้าฝึกอบรบด้วยอัลกอริธึม random forest classifier พบว่า Elite keywords มีค่า Test accuracy สูงที่สุดอยู่ที่ 62.60% และ One-hot encoding มีค่า Test accuracy ต่ำที่สุดอยู่ที่ 54.55% ดังตาราง 15

ตาราง 15 การเปรียบเทียบประสิทธิภาพของ bag-of-words ที่แตกต่างกัน

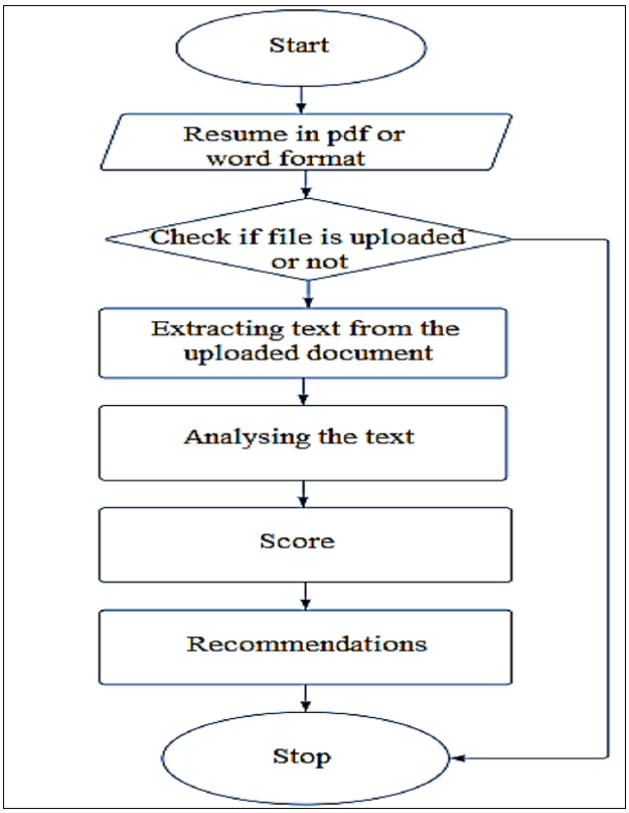
Method	Test accuracy
One-hot encoding	54.55%
TF-IDF	55.36%
TF	58.98%
Elite keywords	62.60%

2.4.10 งานวิจัยเรื่อง A Case Study using RNN-based Keyword Extraction (Sruthi et al., 2023)

งานวิจัยนี้เกิดขึ้นในปี 2023 จุดประสงค์เพื่อระบุผู้สมัครที่มีคุณสมบัติเหมาะสมที่สุดสำหรับตำแหน่งที่เปิดรับสมัครจากข้อมูลที่เป็นประโยชน์ในประวัติ เช่น การศึกษา ทักษะ ความสำเร็จ ประสบการณ์การทำงาน และอื่น ๆ ที่ถูกสร้างขึ้นโดยอัตโนมัติจากเครื่องอัจริยะและการเรียนรู้ของเครื่อง และเพื่อลดระยะเวลาในการสรรหาบุคลากรและลดอคติระหว่างกระบวนการ

คัดเลือก โดยคัดเลือกจากการจัดอันดับผู้สมัครตามข้อมูลในประวัติย่อ ผู้วิจัยจึงได้นำเสนอการพัฒนาเครื่องมือวิเคราะห์ประวัติย่ออัจฉริยะที่สามารถช่วยวิเคราะห์แนะนำคุณสมบัติและทักษะของผู้หางานได้อย่างชาญฉลาด ช่วยเพิ่มประสิทธิภาพและปรับปรุงประวัติย่อของผู้หางานทำให้น่าสนใจยิ่งขึ้นด้วยการใช้เทคโนโลยี การประมวลผลภาษาธรรมชาติและการเรียนรู้ของเครื่อง

เนื่องจาก NLP มีความสามารถในการเข้าใจและแยกวิเคราะห์ข้อมูลในประวัติย่อและดึงข้อมูลที่ต้องการออกมาได้อย่างมีประสิทธิภาพ ด้วยการใช้ไลบรารี Python เช่น NLTK, spaCy, streamlit, pymysql เป็นต้น อีกทั้งยังจัดทำหน้าเว็บไซต์ให้ผู้หางานใช้งานได้อย่างสะดวก โดยระบบสามารถรองรับไฟล์ประวัติย่อในรูปแบบ pdf หรือ word ขนาดไม่เกิน 200MB อนุญาติให้ผู้หางานอัปโหลดไฟล์ประวัติย่อของตนเข้ามาในระบบเพื่อเข้าสู่กระบวนการวิเคราะห์ ดึงข้อมูลออกจากไฟล์ประวัติย่อด้วยการใช้โมดูล spaCy ฟังก์ชัน pyrespareser และ NLTK จะแยกชื่อ อีเมล หมายเลขโทรศัพท์ ทักษะ ประสบการณ์ทั้งหมดของผู้ใช้ สามารถศึกษาแผนผังการทำงานดังภาพประกอบ 26



ภาพประกอบ 26 แผนภาพสถาปัตยกรรม

2.4.11 งานวิจัยเรื่อง Domain Adaptation for Resume Classification Using Convolutional Neural Networks (Sayfullina et al., 2017)

งานวิจัยนี้เกิดขึ้นในปี 2017 นำเสนอวิธีการในการจัดประเภทข้อมูลประวัติย่อของผู้สมัครงานโดยเปรียบเทียบการประเมินผล 2 วิธี ได้แก่

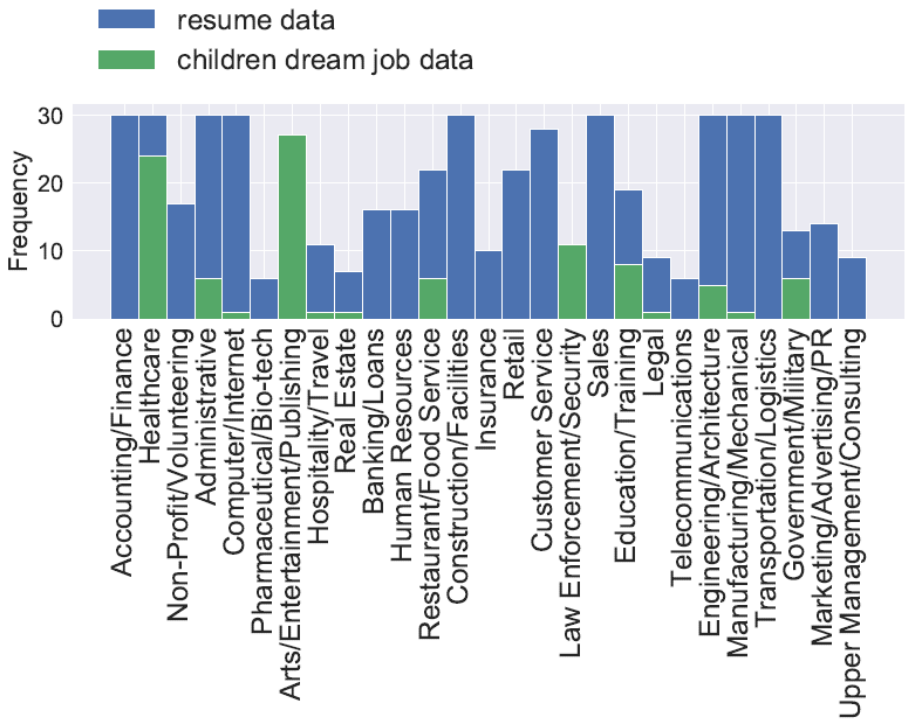
- Fast text Classier
- โครงข่ายประสาทแบบคอนโวลูชัน (Convolutional Neural Networks: CNN)

โดยศึกษากลุ่มข้อมูล 3 ชุดที่แตกต่างกัน ได้แก่

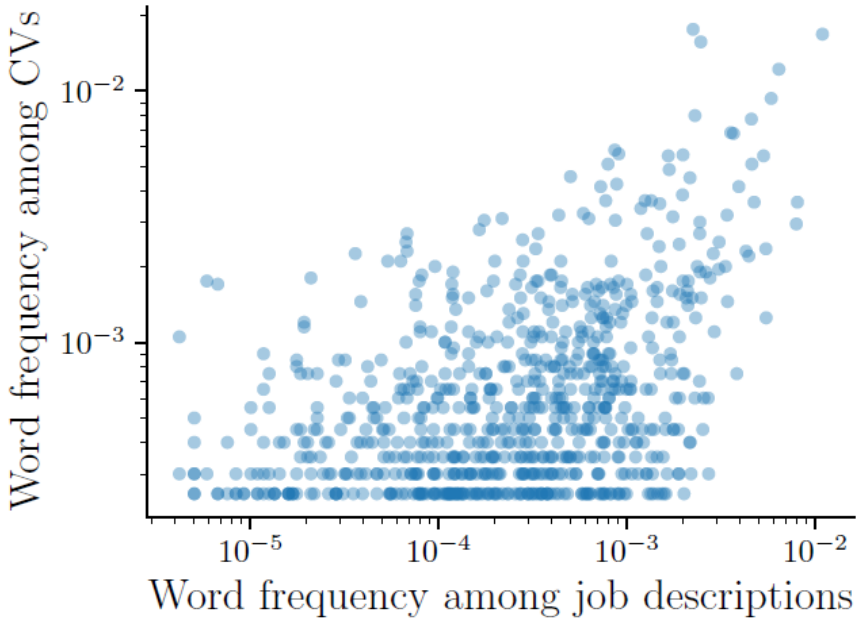
- Job Descriptions ใช้สำหรับฝึกอบรมแบบจำลอง (Train dataset) ซึ่งรวบรวมตำแหน่งงานจาก Indeed Job Search API (https://www.indeed.com/find-jobs.jsp) กว่า 90,000 รายการแย่งประเภทงานออกเป็น 27 สายงานที่แตกต่างกัน ชนิดข้อมูลในส่วนนี้เป็นแบบ unstructured data
- Resume ใช้สำหรับทดสอบแบบจำลอง (Test dataset) โดยรวบรวมตัวอย่างข้อมูลประวัติย่อที่ไม่ระบุตัวตนจำนวน 523 ตัวอย่าง แต่ละตัวอย่างมีป้ายกำกับหนึ่งหมวดหมู่ จาก 27 หมวดหมู่ตามประเภทงานที่ผู้สมัครกำลังมองหา
- Children's Dream Jobs ค่อนข้างเป็นข้อมูลที่ขาดตัวอย่างที่เพียงพอสำหรับการจำแนกประเภทงาน รายละเอียดงานเหล่านี้แตกต่างกันอย่างเห็นได้ชัด ดังนั้นจึงน่าสนใจที่จะนำมาทดลอง มีจำนวนทั้งหมด 98 รายการ ที่ถูกจัดหมวดหมู่แบบไม่อัตโนมัติตาม 27 สายงาน ดังภาพประกอบ 27

1. Accounting/Finance	10. Banking/Loans	19. Education/Training
2. Healthcare	11. Human Resources	20. Legal
3. Non-Profit/Volunteering	12. Restaurant/Food service	21. Telecommunications
4. Administrative	13. Construction/Facilities	22. Engineering/Architecture
5. Computer/Internet	14. Insurance	23. Manufacturing/Mechanical
6. Pharmaceutical/Bio-tech	15. Retail	24. Transportation/Logistics
7. Arts/Entertainment/Publishing	16. Customer Service	25.Government/Military
8. Hospitality/Travel	17. Law Enforcement/Security	26. Marketing/Advertising/PR
9. Real Estate	18. Sales	27. Upper Management/Consulting

ภาพประกอบ 27 ประเภทงาน 27 ประเภท



ภาพประกอบ 28 เปรียบเทียบข้อมูลของ *Resume* กับ *Children's Dream Jobs*



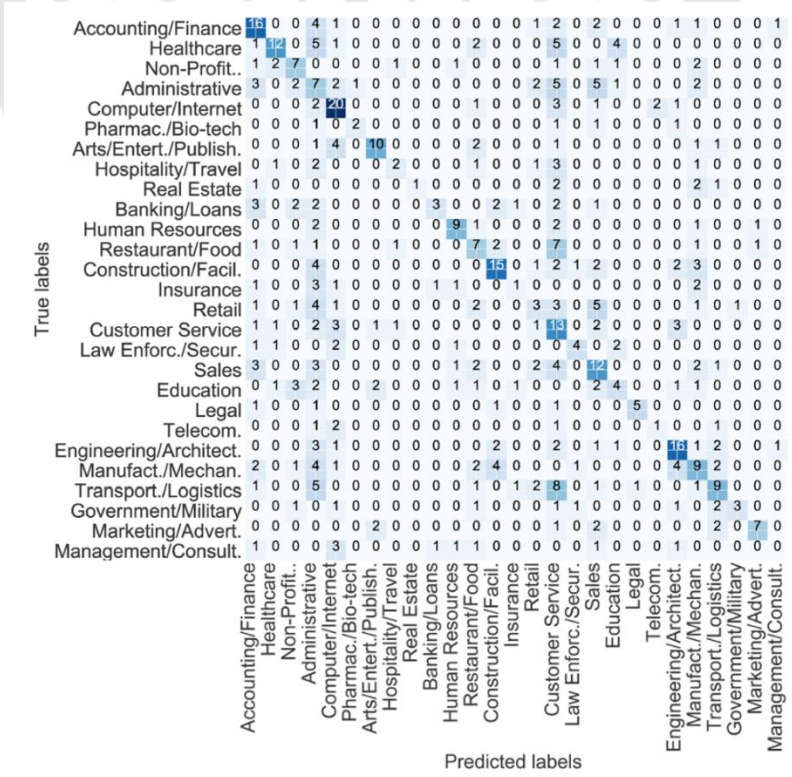
ภาพประกอบ 29 เปรียบเทียบคำที่มีพบในชุดข้อมูล *Job Descriptions* กับ *Resume*

จากภาพประกอบ 28, 29 พบว่าข้อมูลใน Resume มักจะใช้คำคุณศัพท์เพื่ออธิบายคุณลักษณะของตนเอง เช่น adaptable, polite ในขณะ Job Descriptions กล่าวถึงบทบาทมากกว่า เช่น director, coordinator

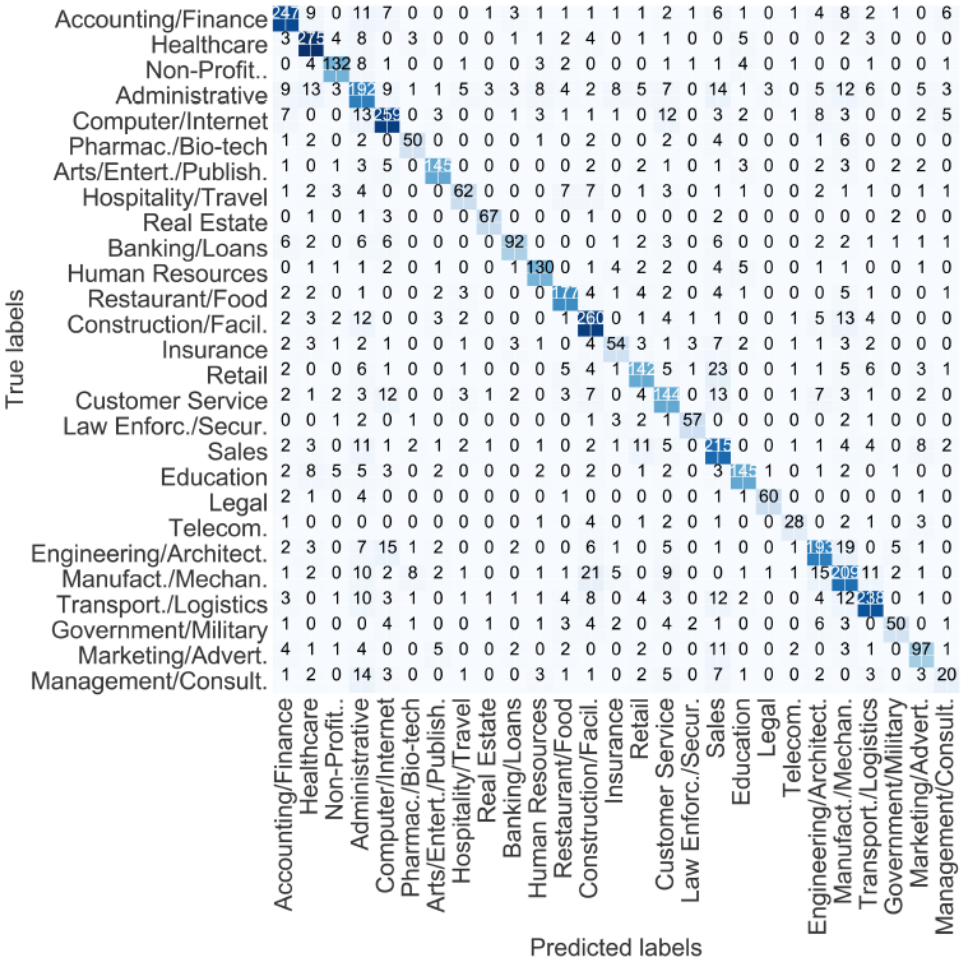
ผู้วิจัยใช้ข้อมูล Training data จำนวน 80,000 ตัวอย่าง และValidation data 5,000 ตัวอย่าง และใช้ข้อมูล Test data จาก Resume และ children's dream job จำนวน 5,000 ตัวอย่าง พบว่า ข้อมูล Job Description มีค่า Accuracy สูงที่สุดทั้ง fastText และ CNN อยู่ที่ 71.99%, 74.88% ตามลำดับ ดังตาราง 16 และภาพประกอบ 30, 31

ตาราง 16 % Accuracy ระหว่าง fastText กับ CNN ในแต่ละชุดข้อมูล

Dataset	FastText	CNN
Job description	71.99	74.88
Resume	33.40	40.15
Children's dream job	28.50	51.02



ภาพประกอบ 30 Confusion Matrix ของชุดข้อมูล Resume



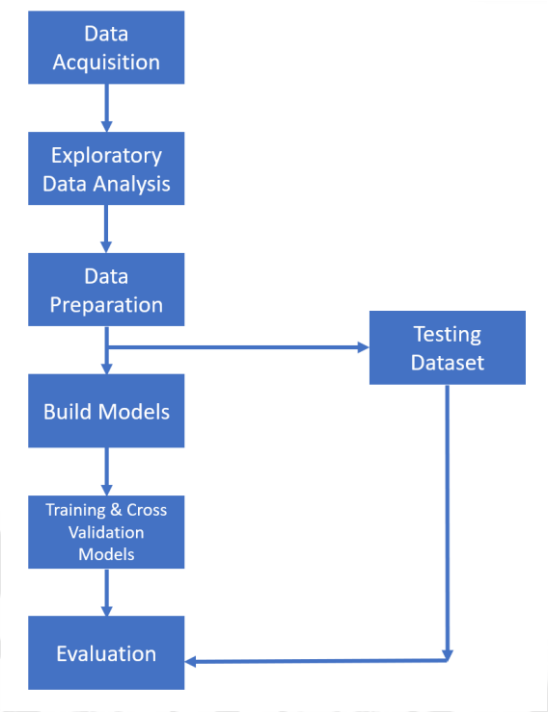
ภาพประกอบ 31 Confusion Matrix ของชุดข้อมูล Job Description

บทที่ 3
กระบวนการ และวิธีการดำเนิการวิจัย

ในการวิจัยครั้งนี้ ผู้วิจัยมุ่งเน้นศึกษาวิธีการคัดกรองผู้สมัครจากประวัติย่อด้วยหลักการเรียนรู้ของเครื่องและประเมินประสิทธิภาพจากแบบจำลอง 8 แบบ ได้แก่

1. Support Vector Classification (SVC)
2. Logistic Regression
3. Random Forest
4. K-Nearest Neighbors
5. Gradient Boosting
6. AdaBoost Classifier
7. Gaussian Naïve Bayes
8. Decision Tree

อีกทั้งยังใช้เทคนิค OneVsRestClassifier เป็นคลาสย่อยของ Classifier ในไลบรารี scikit-learn ที่ใช้เพื่อจำแนกประเภทแบบ multi-class โดยสร้างแบบจำลองย่อยแยกกันสำหรับแต่ละคลาส โดยใช้การวัดผลด้วยการดูค่า Accuracy, Precision, Recall, F1-score ควบคู่กับทำ Cross Validation แบบ 10 fold เพื่อค้นหาแบบจำลองที่เหมาะสมในการคัดกรองทักษะของผู้สมัครให้ตรงกับตำแหน่งงาน ซึ่งจากปัจจุบันผู้สรรหาต้องคัดเลือกผู้สมัครโดยอ่านรายละเอียดในประวัติย่อทีละฉบับ มาเป็นการทำงานแบบอัตโนมัติใช้ปัญญาประดิษฐ์ช่วยวิเคราะห์ เพื่อช่วยเพิ่มประสิทธิภาพในการสรรหาบุคลากร และประหยัดเวลาในการทำงานของผู้สรรหา อีกทั้งยังช่วยลดความผิดพลาดที่เกิดจากมนุษย์ และท้ายที่สุดช่วยลดอัตราการว่างงานได้อีกด้วย ขั้นตอนการดำเนินงานวิจัยแบ่งออกเป็น 6 ขั้นตอน ดังภาพประกอบ 32



ภาพประกอบ 32 ขั้นตอนการทำงานวิจัย

3.1 **การรวบรวมข้อมูล** (Data Acquisition)

เนื่องจากข้อมูลประวัติย่อ เป็นข้อมูลที่มีความเป็นส่วนตัว และละเอียดอ่อน จึงทำให้ยากต่อการหาชุดข้อมูลจริงของผู้สมัคร ดังนั้นผู้วิจัยจึงใช้ข้อมูลจาก kaggle.com ซึ่งเป็น Public Dataset ชื่อ Updated Resume Dataset มาใช้ในงานวิจัยนี้ ซึ่งในชุดข้อมูลประกอบด้วย ประเภทงาน (Category) และ ประวัติย่อ (Resume) (Dutta, 2021) ดังภาพประกอบ 33

	Category	Resume
0	Data Science	Skills * Programming Languages: Python (pandas...
1	Data Science	Education Details \r\nMay 2013 to May 2017 B.E...
2	Data Science	Areas of Interest Deep Learning, Control Syste...
3	Data Science	Skills â□¢ R â□¢ Python â□¢ SAP HANA â□¢ Table...
4	Data Science	Education Details \r\n MCA YMCAUST, Faridab...
...	...	...
957	Testing	Computer Skills: â□¢ Proficient in MS office (...
958	Testing	â□□ Willingness to accept the challenges. â□□ ...
959	Testing	PERSONAL SKILLS â□¢ Quick learner, â□¢ Eagerne...
960	Testing	COMPUTER SKILLS & SOFTWARE KNOWLEDGE MS-Power ...
961	Testing	Skill Set OS Windows XP/7/8/8.1/10 Database MY...

962 rows × 2 columns

ภาพประกอบ 33 ชุดข้อมูลที่ใช้ในงานวิจัยนี้

3.2 **การวิเคราะห์ข้อมูลเชิงสำรวจ** (Exploratory Data Analysis: EDA)

เมื่อนำข้อมูลเข้ามาแล้ว ขั้นตอนต่อไปคือการทำความเข้าใจกับข้อมูล ซึ่งพบว่าข้อมูลมีขนาด 962 แถว 2 คอลัมน์ ชนิดข้อมูลแบบ objective และไม่มีค่าว่าง (Missing Value) ดังภาพประกอบ 34 และมีจำนวนประเภทงานที่ไม่ซ้ำกันทั้งหมด 25 ประเภท ดังภาพประกอบ 35 ได้แก่ ประเภทงานตามภาพประกอบ 36

```
 1 df.shape
```

```
(962, 2)
```

```
 1 df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 962 entries, 0 to 961
Data columns (total 2 columns):
 #   Column    Non-Null Count  Dtype 
---  ------    --------------  ----- 
 0   Category  962 non-null    object
 1   Resume    962 non-null    object
dtypes: object(2)
memory usage: 15.2+ KB
```

```
 1 print(df.isnull().sum())
```

```
Category    0
Resume      0
dtype: int64
```

ภาพประกอบ 34 ตรวจสอบขนาดข้อมูล ชนิดข้อมูล และค่าว่าง

```
[ ]  1 #The nunique() function counts the number of unique entries in a column of a dataframe.
     2 df['Category'].nunique()
```

```
25
```

ภาพประกอบ 35 แสดงจำนวนประเภทงานที่ไม่ซ้ำกัน

```
1 for i in range(len(df['Category'].unique())):
2     print(df['Category'].unique()[i])
```

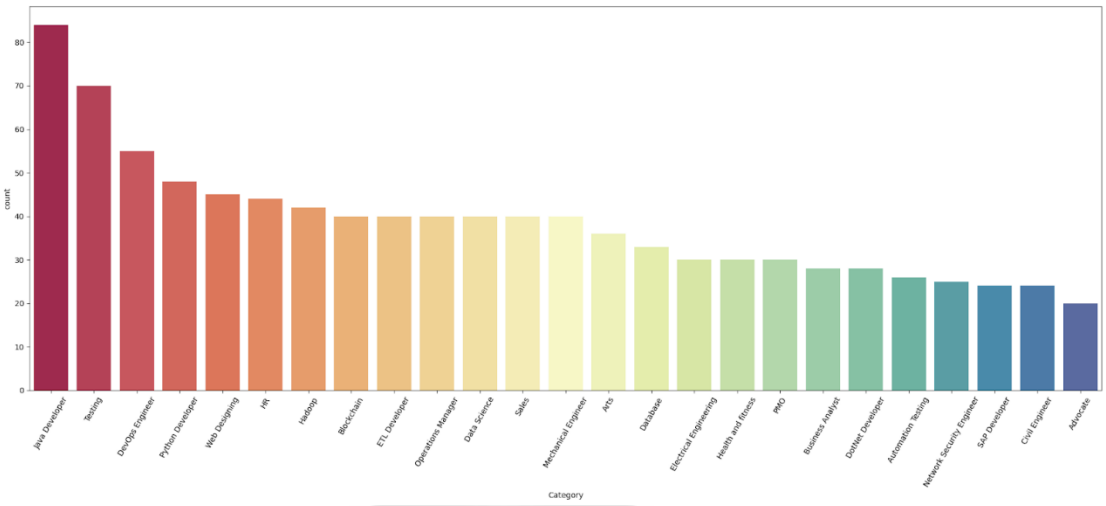
```
Data Science
HR
Advocate
Arts
Web Designing
Mechanical Engineer
Sales
Health and fitness
Civil Engineer
Java Developer
Business Analyst
SAP Developer
Automation Testing
Electrical Engineering
Operations Manager
Python Developer
DevOps Engineer
Network Security Engineer
PMO
Database
Hadoop
ETL Developer
DotNet Developer
Blockchain
Testing
```

ภาพประกอบ 36 แสดงชื่อประเภทงานที่ไม่ซ้ำกัน

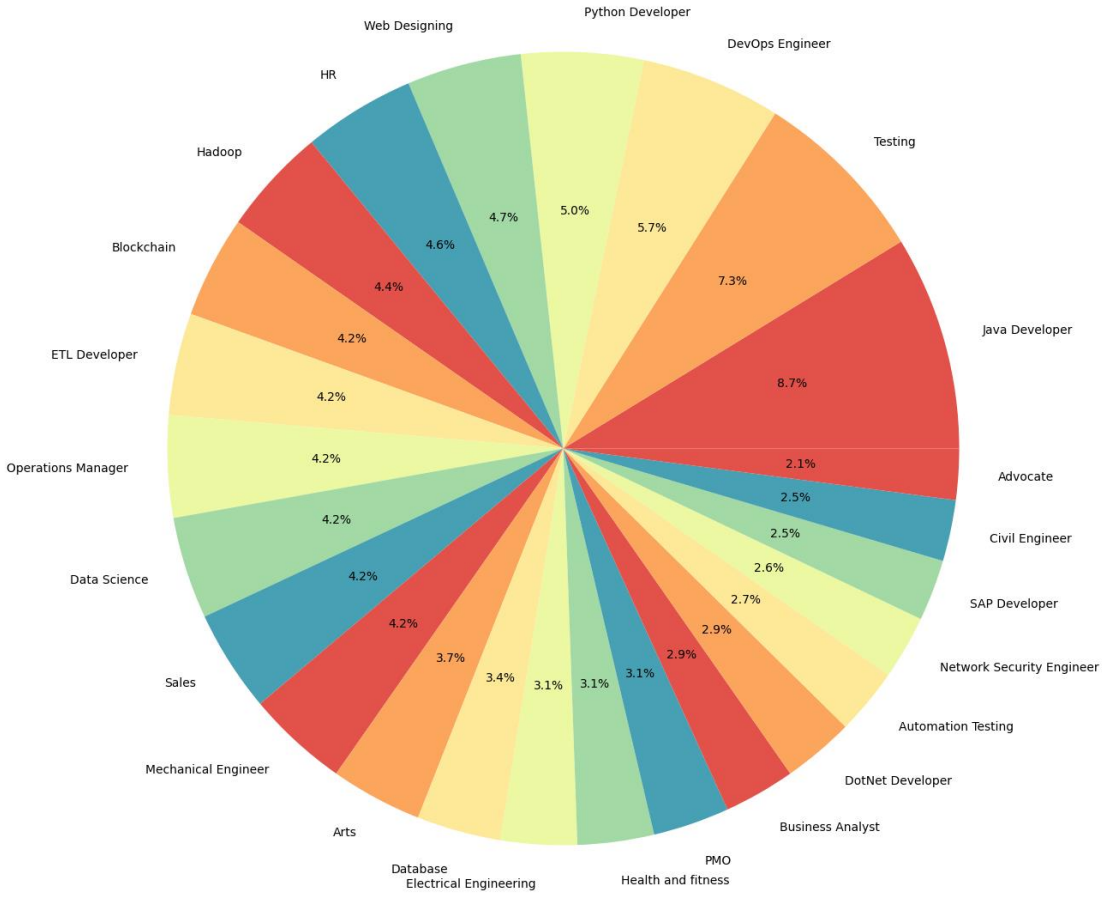
	index	Category
0	Java Developer	84
1	Testing	70
2	DevOps Engineer	55
3	Python Developer	48
4	Web Designing	45
5	HR	44
6	Hadoop	42
7	Blockchain	40
8	ETL Developer	40
9	Operations Manager	40
10	Data Science	40
11	Sales	40
12	Mechanical Engineer	40
13	Arts	36
14	Database	33
15	Electrical Engineering	30
16	Health and fitness	30
17	PMO	30
18	Business Analyst	28
19	DotNet Developer	28
20	Automation Testing	26
21	Network Security Engineer	25
22	SAP Developer	24
23	Civil Engineer	24
24	Advocate	20

ภาพประกอบ 37 แสดงจำนวนประวัติย่อของแต่ละประเภทงาน

จากภาพประกอบ 37-39 พบว่า 3 อันดับประเภทงานที่มีประวัติย่อมากที่สุด ได้แก่ 1. Java Developer (84 ประวัติย่อ) 2. Testing (70 ประวัติย่อ) 3. DevOps Engineer (55 ประวัติย่อ)



ภาพประกอบ 38 แผนภูมิแท่งแสดงจำนวนประวัติย่อของแต่ละประเภทงาน



ภาพประกอบ 39 แผนภูมิวงกลมแสดงจำนวนประวัติย่อของแต่ละประเภทงาน

แสดงคำที่ใช้บ่อยที่สุดของประวัติย่อในแต่ละประเภทงาน ดังภาพประกอบ 40- 64

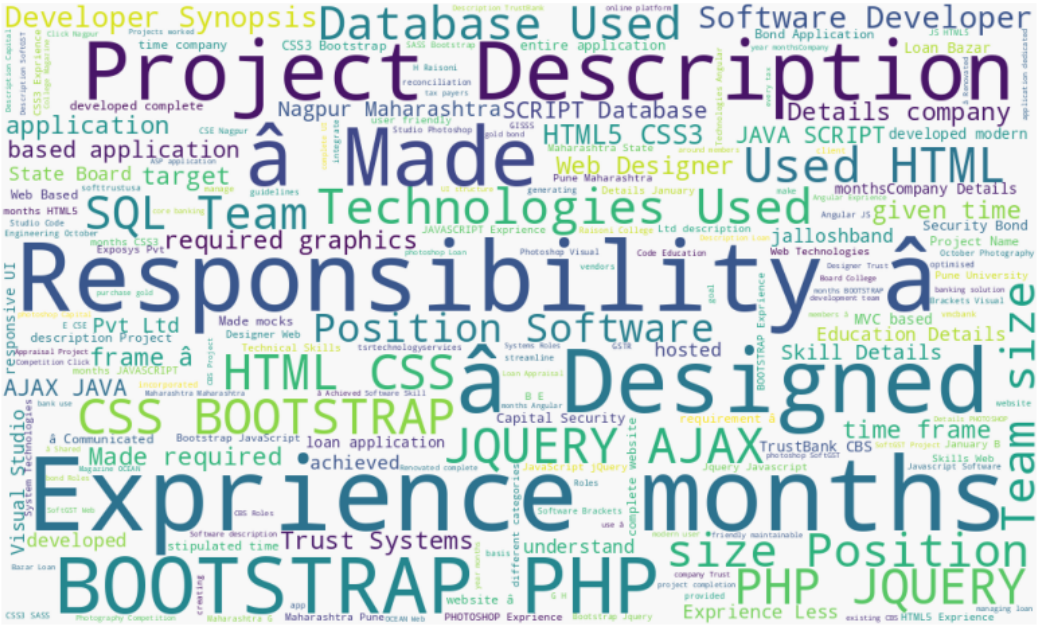
ภาพประกอบ 40 คำที่ใช้บ่อยที่สุดใน resume ของประเภทงาน Data Science

ภาพประกอบ 41 คำที่ใช้บ่อยที่สุดใน resume ของประเภทงาน HR

ภาพประกอบ 42 คำที่ใช้บ่อยที่สุดใน *resume* ของประเภทงาน *Advocate*

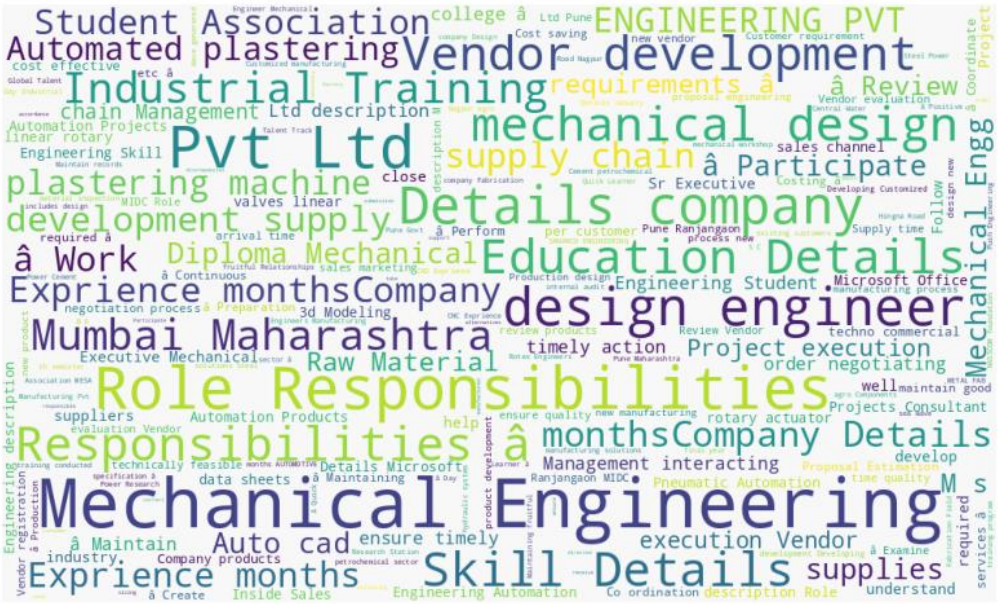
ภาพประกอบ 43 คำที่ใช้บ่อยที่สุดใน *resume* ของประเภทงาน *Art*

Words Commonly Used in *WebDesigning* Resumes



ภาพประกอบ 44 คำที่ใช้บ่อยที่สุดใน *resume* ของประเภทงาน *Web Designing*

Words Commonly Used in *MechanicalEngineer* Resumes



ภาพประกอบ 45 คำที่ใช้บ่อยที่สุดใน *resume* ของประเภทงาน *Mechanical Engineer*

ภาพประกอบ 46 คำที่ใช้บ่อยที่สุดใน *resume* ของประเภทงาน *Sales*

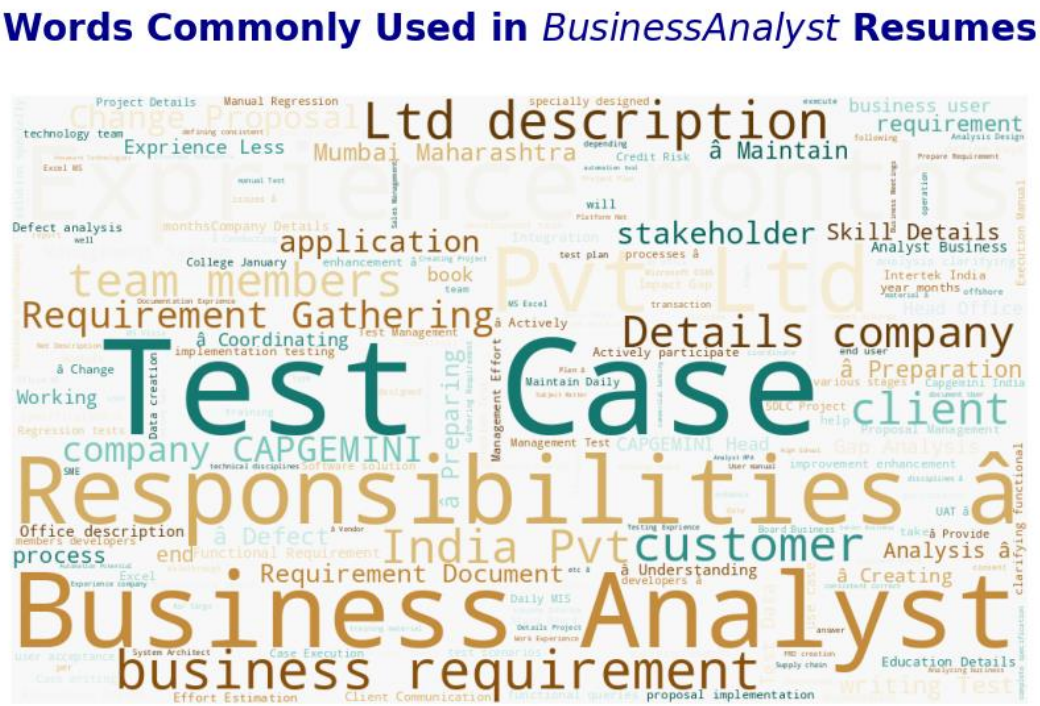
ภาพประกอบ 47 คำที่ใช้บ่อยที่สุดใน *resume* ของประเภทงาน *Health and fitness*

Words Commonly Used in *CivilEngineer* Resumes

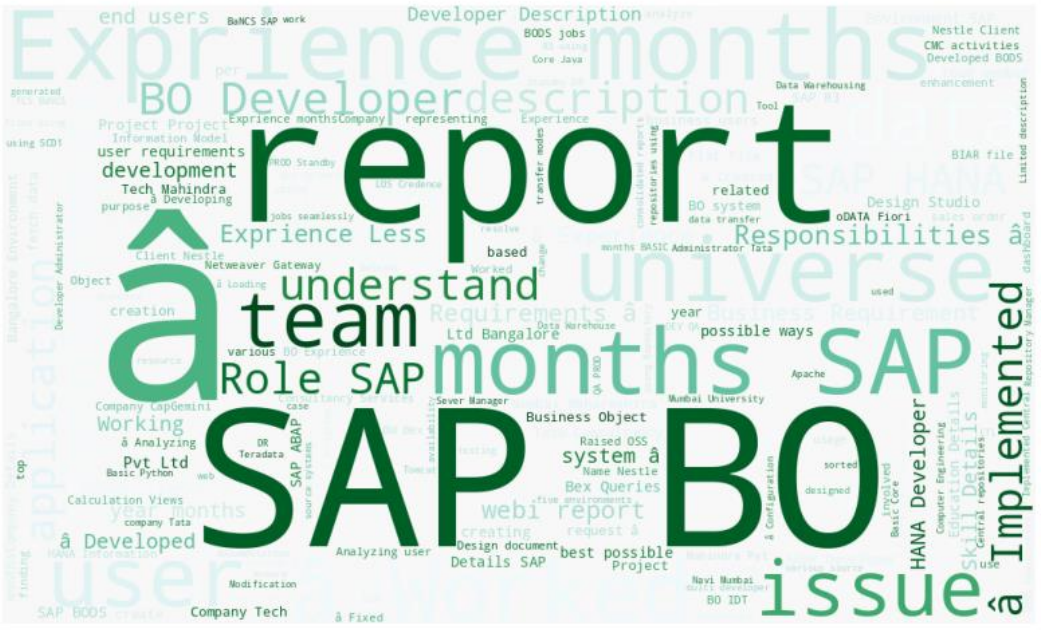
ภาพประกอบ 48 คำที่ใช้บ่อยที่สุดใน *resume* ของประเภทงาน *Civil Engineer*

Words Commonly Used in *JavaDeveloper* Resumes

ภาพประกอบ 49 คำที่ใช้บ่อยที่สุดใน *resume* ของประเภทงาน *Java Developer*



ภาพประกอบ 50 คำที่ใช้บ่อยที่สุดใน *resume* ของประเภทงาน *Business Analyst*

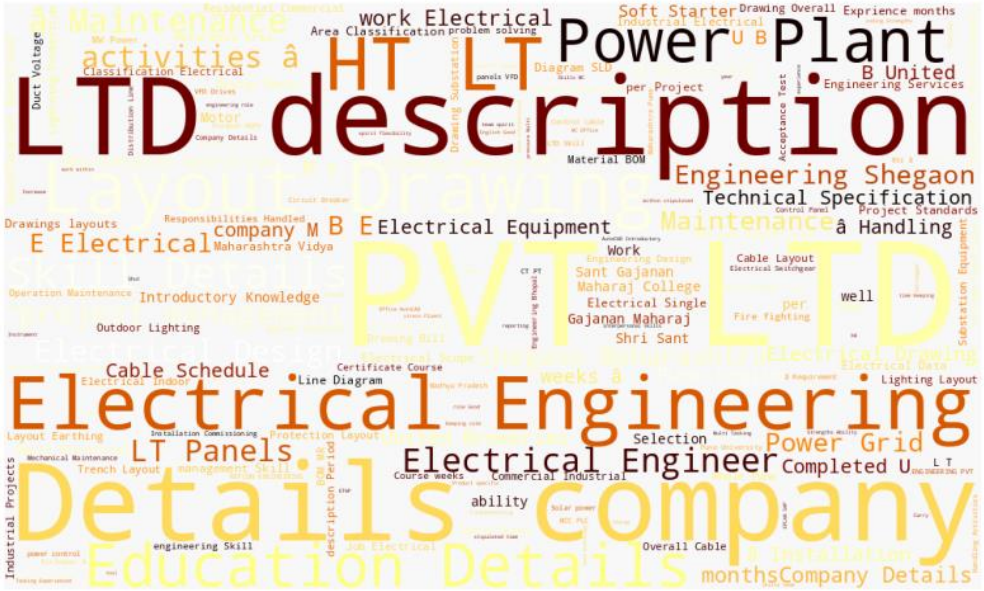


ภาพประกอบ 51 คำที่ใช้บ่อยที่สุดใน *resume* ของประเภทงาน *SAP Developer*

Words Commonly Used in *AutomationTesting* Resumes

ภาพประกอบ 52 คำที่ใช้บ่อยที่สุดใน *resume* ของประเภทงาน *Automation Testing*

Words Commonly Used in *ElectricalEngineering* Resumes



ภาพประกอบ 53 คำที่ใช้บ่อยที่สุดใน *resume* ของประเภทงาน *Electrical Engineering*

ภาพประกอบ 54 คำที่ใช้บ่อยที่สุดใน *resume* ของประเภทงาน *Operations Manager*

ภาพประกอบ 55 คำที่ใช้บ่อยที่สุดใน *resume* ของประเภทงาน *Python Developer*

Words Commonly Used in *DevOpsEngineer* Resumes

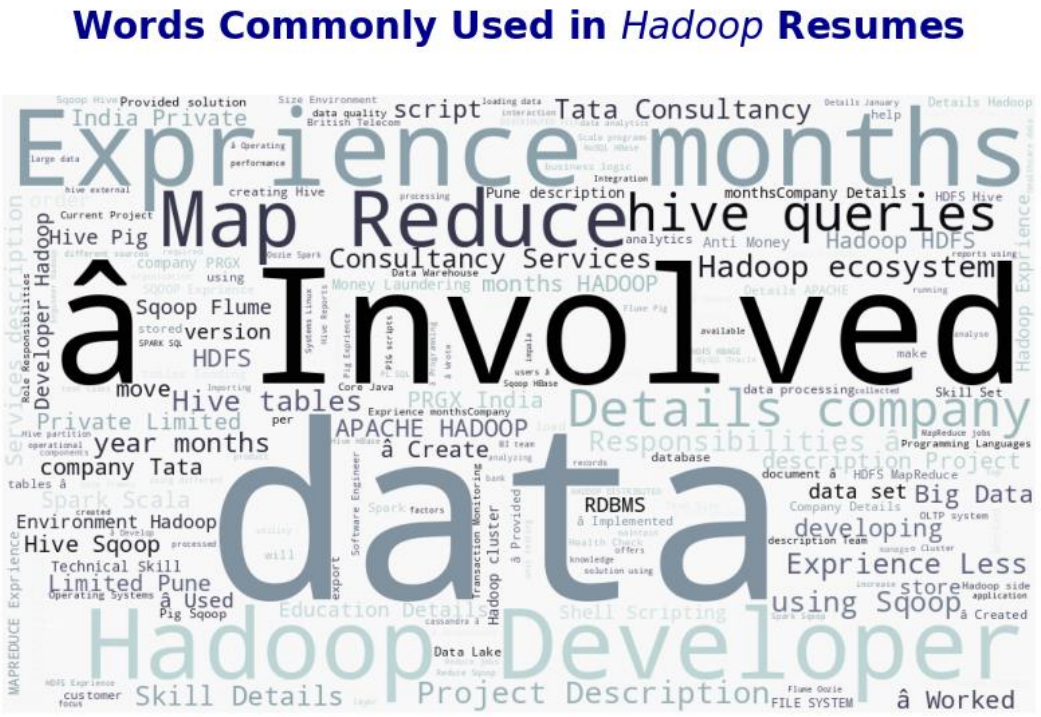
ภาพประกอบ 56 คำที่ใช้บ่อยที่สุดใน *resume* ของประเภทงาน *DevOps Engineer*

Words Commonly Used in *NetworkSecurityEngineer* Resumes

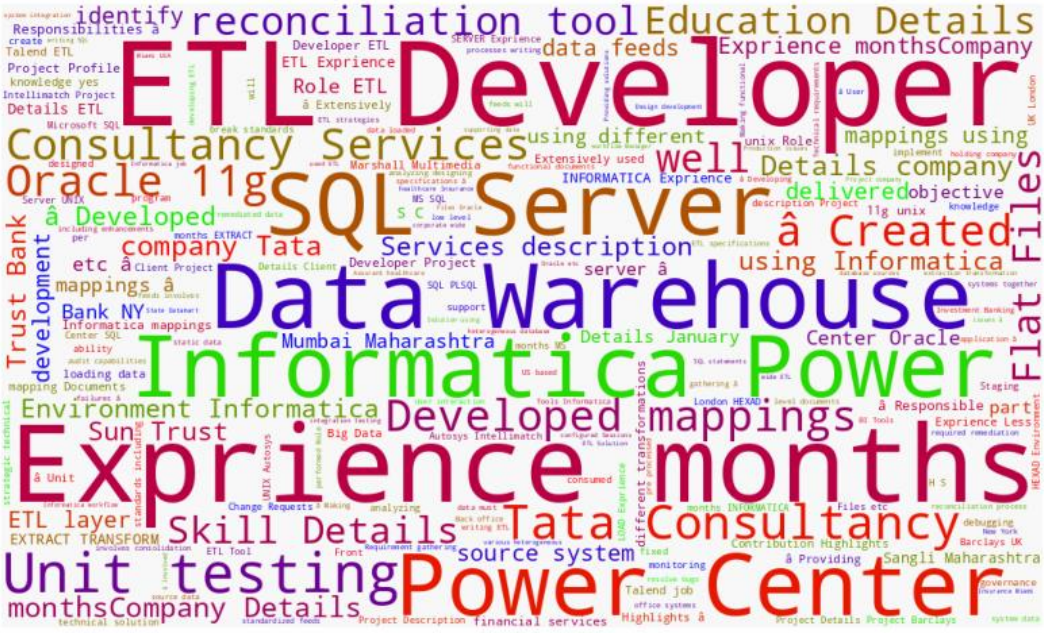
ภาพประกอบ 57 คำที่ใช้บ่อยที่สุดใน *resume* ของประเภทงาน *Network Security Engineer*

ภาพประกอบ 58 คำที่ใช้บ่อยที่สุดใน *resume* ของประเภทงาน *PMO*

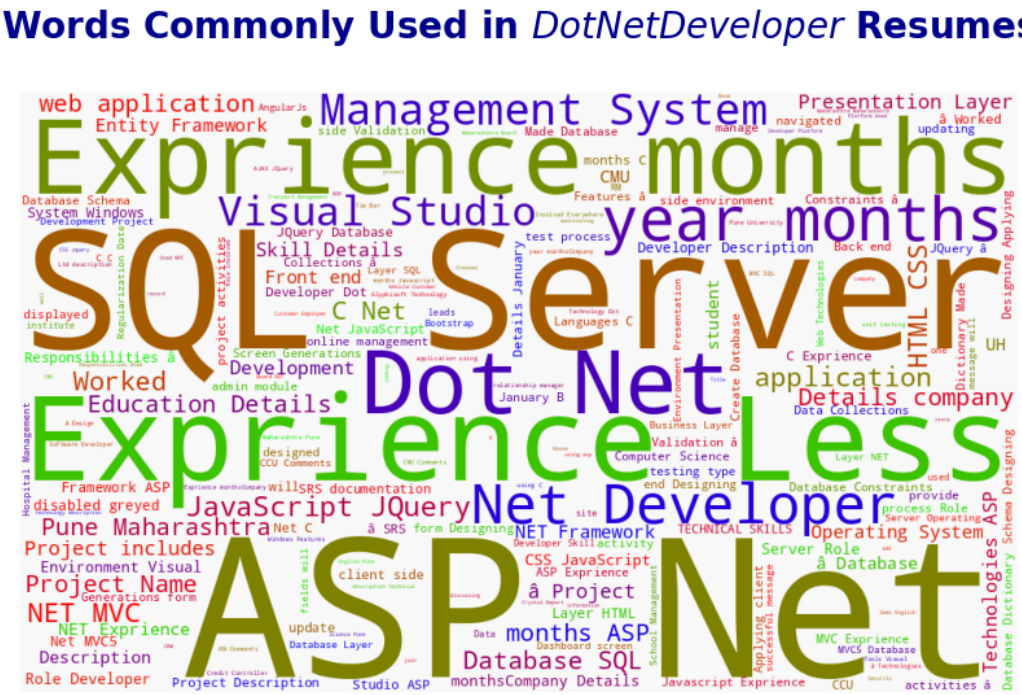
ภาพประกอบ 59 คำที่ใช้บ่อยที่สุดใน *resume* ของประเภทงาน *Database*



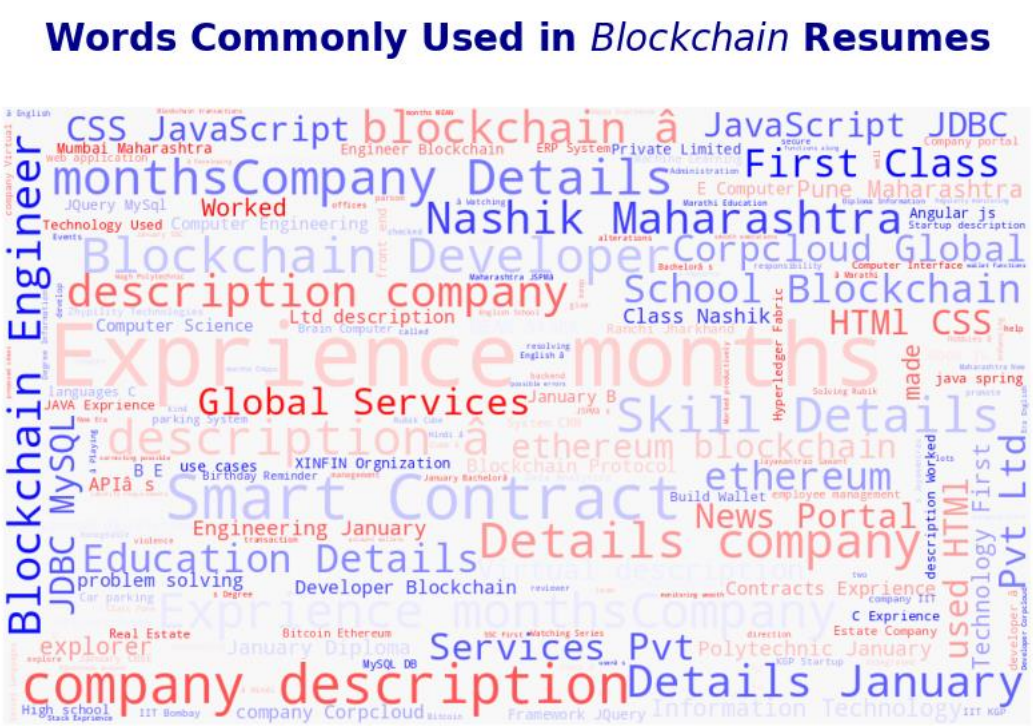
ภาพประกอบ 60 คำที่ใช้บ่อยที่สุดใน *resume* ของประเภทงาน *Hadoop*



ภาพประกอบ 61 คำที่ใช้บ่อยที่สุดใน *resume* ของประเภทงาน *ETL Developer*



ภาพประกอบ 62 คำที่ใช้บ่อยที่สุดใน *resume* ของประเภทงาน *DotNet Developer*



ภาพประกอบ 63 คำที่ใช้บ่อยที่สุดใน *resume* ของประเภทงาน *Blockchain*

ภาพประกอบ 64 คำที่ใช้บ่อยที่สุดใน *resume* ของประเภทงาน *Testing*

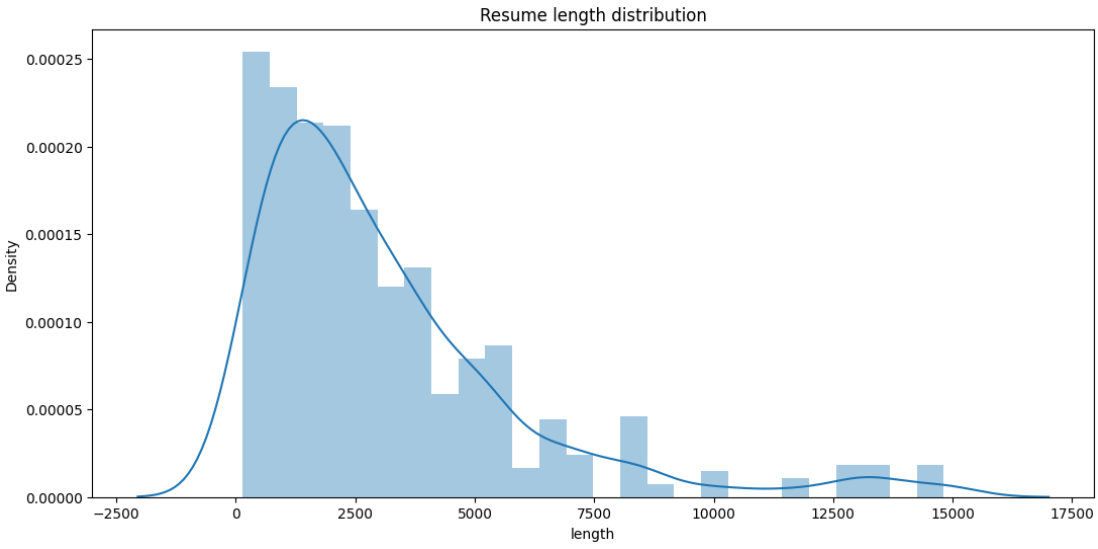
แสดงสถิติเชิงบรรยายทางคณิตศาสตร์ เพื่อดูจำนวนตัวอักษรในประวัติย่อ และการกระจายตัว ดังภาพประกอบ 65-67 ตามลำดับ มีรายละเอียดดังนี้

- count หมายถึง จำนวนข้อมูลทั้งหมดในชุดข้อมูลนี้คือ 962 รายการ
- mean หมายถึง ค่าเฉลี่ยของความยาวประวัติย่อคือ 3,160.36 คำ
- std หมายถึง ค่าเบี่ยงเบนมาตรฐานของความยาวประวัติย่อคือ 2,886.53 คำ
- min หมายถึง ความยาวประวัติย่อที่สั้นที่สุดคือ 142 คำ
- 25% หมายถึง 25% ของประวัติย่อ ยาวน้อยกว่าหรือเท่ากับ 1,217 คำ
- 50% หมายถึง 50% ของประวัติย่อ ยาวน้อยกว่าหรือเท่ากับ 2,355 คำ
- 75% หมายถึง 75% ของประวัติย่อ ยาวน้อยกว่าหรือเท่ากับ 4,073 คำ
- max หมายถึง ความยาวประวัติย่อ ที่สุดคือ 14,816 คำ

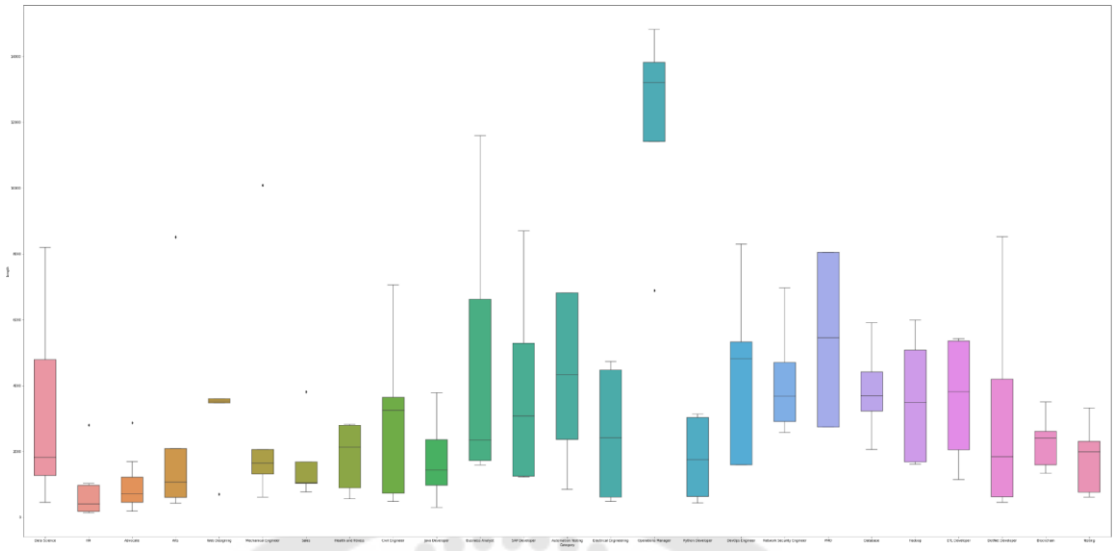
```
1 df['length'].describe()
```

```
count      962.000000
mean      3160.364865
std       2886.528521
min        142.000000
25%       1217.250000
50%       2355.000000
75%       4073.750000
max      14816.000000
Name: length, dtype: float64
```

ภาพประกอบ 65 สถิติเชิงบรรยายทางคณิตศาสตร์ของตัวอักษรในประวัติย่อ



ภาพประกอบ 66 กราฟแสดงการกระจายตัวของตัวอักษรในประวัติย่อ



ภาพประกอบ 67 แผนภูมิกล่องแสดงจำนวนคำในแต่ละประเภทของประวัติย่อ

3.3 **การเตรียมข้อมูล** (Data Preparation)

ในขั้นตอนการเตรียมข้อมูล ผู้วิจัยได้ใช้ชุดข้อมูลของประวัติย่อ และประเภทงานที่ผ่านการทำ labelling เรียบร้อยแล้ว จากนั้นนำชุดข้อมูลนี้มาเข้าสู่กระบวนการประมวลผลภาษาธรรมชาติ (NLP) โดยมีขั้นตอนดังต่อไปนี้

3.3.1 ลบคำต่าง ๆ ที่ไม่สำคัญในประวัติย่อออก เช่น URLs, RT, cc, hashtags, @, ตัวอักษรพิเศษ และช่องว่าง ผลลัพธ์ดังภาพประกอบ 68

	Category	Resume	cleaned_resume
0	Data Science	Skills * Programming Languages: Python (pandas...	Skills Programming Languages Python pandas num...
1	Data Science	Education Details \r\nMay 2013 to May 2017 B.E...	Education Details May 2013 to May 2017 B E UIT...
2	Data Science	Areas of Interest Deep Learning, Control Syste...	Areas of Interest Deep Learning Control System...
3	Data Science	Skills â□¢ R â□¢ Python â□¢ SAP HANA â□¢ Table...	Skills R Python SAP HANA Tableau SAP HANA SQL ...
4	Data Science	Education Details \r\n MCA YMCAUST, Faridab...	Education Details MCA YMCAUST Faridabad Haryan...

ภาพประกอบ 68 แสดงข้อมูลหลังลบคำที่ไม่สำคัญ และตัวอักษรพิเศษออก

3.3.2 แปลงข้อมูลใน Category จากข้อความให้เป็นข้อมูลตัวเลข โดยใช้คำสั่ง Label Encoding เพื่อให้แต่ละหมวดหมู่กลายเป็นคลาส ก่อนจะสร้างแบบจำลองการจำแนกประเภทหลายคลาส (multiclass classification model)

3.3.3 Tokenization คือกระบวนการแบ่งข้อความออกเป็นหน่วยเล็ก ๆ ขั้นตอนนี้มีความสำคัญเนื่องจากจะแบ่งข้อมูลออกเป็นหน่วยขนาดเล็กที่ใช้งานได้และง่ายต่อการประมวลผล หน่วยข้อความขนาดเล็กเหล่านี้เรียกว่าโทเค็น โทเค็นเหล่านี้สามารถช่วยในการทำความเข้าใจบริบทของข้อความและในการสร้างแบบจำลอง NLP จากนั้น ตัด stop words หรือ คำที่เจอบ่อย ๆ แต่ไม่สื่อความหมายออก เช่น ‘nor’, ‘me’, ‘were’, ‘her’, ‘more’, ‘himself’, ‘this’ ข้อดีคือช่วยลดปริมาณข้อมูลที่ต้องประมวลผล ทำให้การทำงานรวดเร็วขึ้น สุดท้ายทำ Lemmatization หรือการเปลี่ยนรูปคำให้อยู่ในรูปแบบของคำดั้งเดิมหรือคำกริยาช่องที่ 1 เพื่อให้อยู่ในรากศัพท์เดียวกัน จำนวนคำที่ทำ Lemmatize แล้วอยู่ที่ 895 คำ ดังภาพประกอบ 69

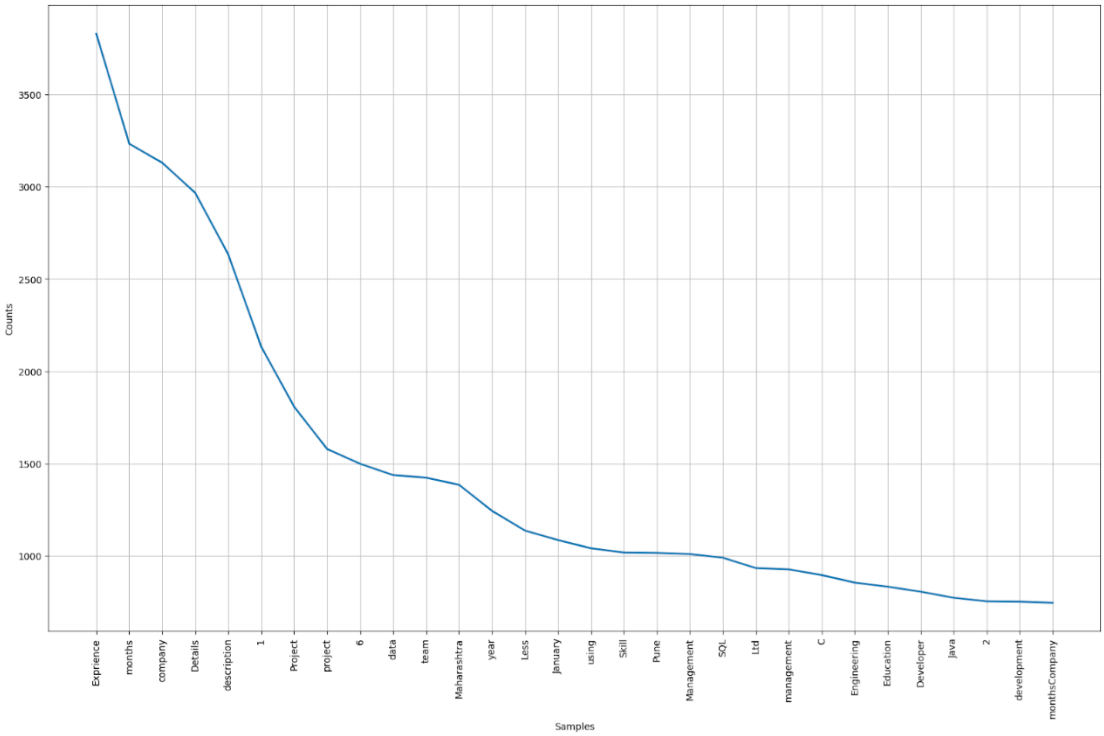
```
Number of words Lemmatized= 895
Number of words not Lemmatized= 9105
```

ภาพประกอบ 69 แสดงจำนวนการทำ *Lemmatize* และไม่ได้ทำ *Lemmatize*

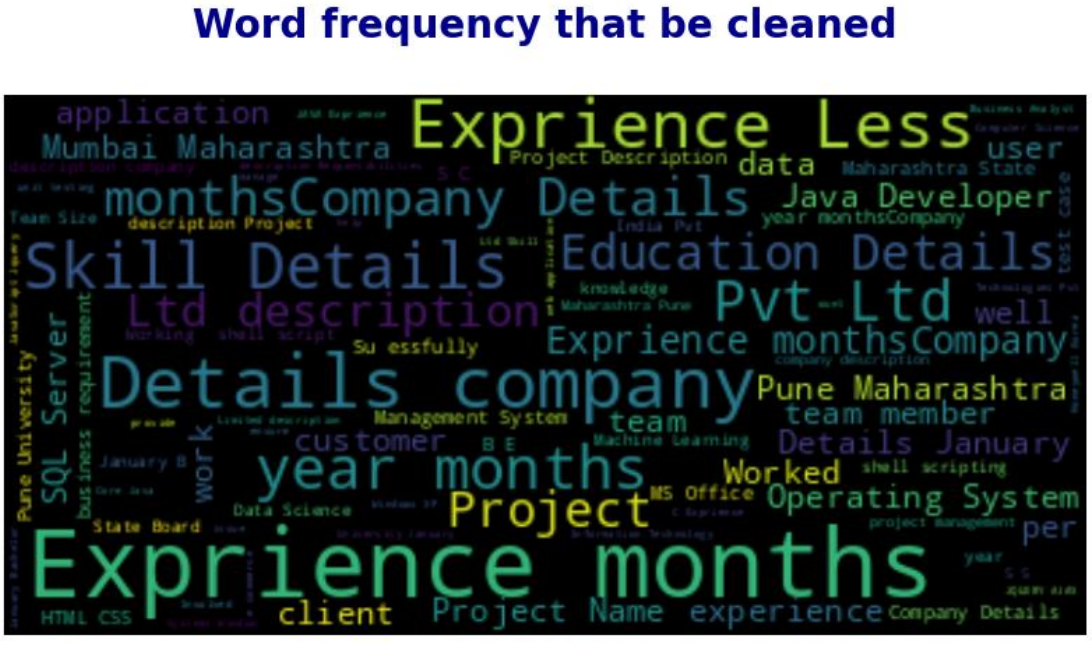
3.3.4 ใช้ฟังก์ชัน FreqDist เพื่อดูความถี่ของคำทั้งหมดในข้อความ ดังภาพประกอบ 70-72 แสดงให้เห็นว่าคำว่า Experience พบมากที่สุดจำนวน 3,829 ครั้ง

```
[('Exprience', 3829),
 ('months', 3233),
 ('company', 3130),
 ('Details', 2967),
 ('description', 2634),
 ('1', 2134),
 ('Project', 1808),
 ('project', 1579),
 ('6', 1499),
 ('data', 1438),
 ('team', 1424),
 ('Maharashtra', 1385),
 ('year', 1244),
 ('Less', 1137),
 ('January', 1086),
 ('using', 1041),
 ('Skill', 1018),
 ('Pune', 1016),
 ('Management', 1010),
 ('SQL', 990)]
```

ภาพประกอบ 70 คำที่พบมากที่สุดในประวัติย่อ *30* คำแรก



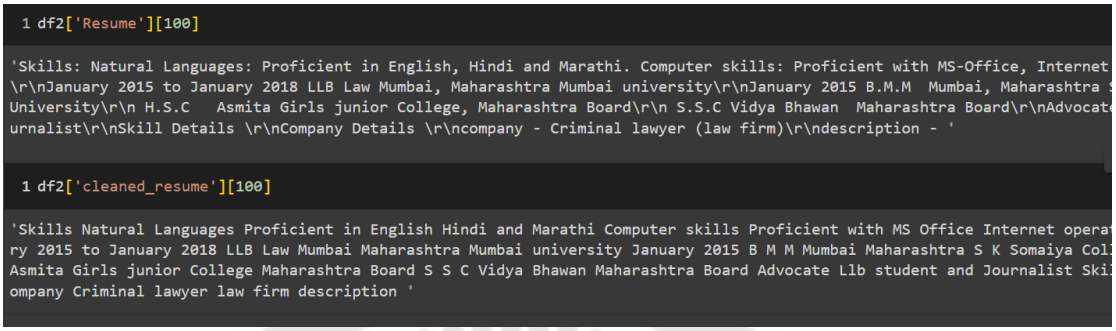
ภาพประกอบ 71 กราฟแสดงคำที่พบมากที่สุดในประวัติย่อ 30 คำแรก



ภาพประกอบ 72 *Word Cloud* คำที่พบมากที่สุดในประวัติย่อ 30 คำแรก

3.4 การสร้างแบบจำลอง (Modeling)

ตรวจสอบข้อมูลให้แน่ใจว่าข้อมูลสะอาด พร้อมนำไปเข้ากระบวนการฝึกอบรม ดังภาพประกอบ 73 จากนั้นแปลงข้อความเป็นเวกเตอร์ตัวเลข ด้วยฟังก์ชัน TF-IDF Vectorization



ภาพประกอบ 73 แสดงข้อความก่อนและหลังทำความสะอาด

TF-IDF หรือ Term Frequency – Inverse Document Frequency เป็นเทคนิคที่พิจารณาองค์ประกอบของคำภายในประโยค (และเอกสาร) เป็นหลักโดยจะไม่นำลำดับของคำภายในเอกสารมาใช้วิเคราะห์ประกอบด้วย มี 2 องค์ประกอบด้วยกันคือ Term Frequency (TF) และ Inverse document Frequency (IDF) ซึ่งการคำนวณค่าของทั้ง TF และ IDF นั้น มีหลายรูปแบบ สิ่งที่เลือกมานำเสนอ ณ ที่นี้จะเป็นรูปแบบพื้นฐานของทั้งสองค่า (แต่การคำนวณในรูปแบบอื่น ๆ นั้นก็จะมีลักษณะคล้ายคลึงกัน)

Term-Frequency (TF) ถ้าหากคำ ๆ ไหนถูกพูดถึงอยู่บ่อย ๆ ในเอกสารนั้น ๆ จะมีความเป็นไปได้สูงว่าคำนั้นมีความเกี่ยวข้องกับใจความสำคัญของเอกสารนั้น ๆ มาก (Prasertsom, 2563) ดังสมการ 1

$$TF(\text{ของคำ ๆ หนึ่ง}) = \frac{\text{จำนวนของคำนั้น ๆ ในเอกสาร}}{\text{จำนวนของคำทั้งหมดในเอกสาร}} \qquad (1)$$

Inverse Document Frequency (IDF) เป็นการคำนวณค่าน้ำหนัก (weight) ความสำคัญของแต่ละคำโดยจะคำที่พบเจอได้บ่อย ๆ (ในหลาย ๆ เอกสาร) จะมีค่า IDF ต่ำ ซึ่งบ่งบอกว่าคำเหล่านั้นจะไม่สามารถดึงเอาจุดเด่นของเอกสารที่คำเหล่านั้นปรากฏอยู่ออกมาได้ดี ค่า IDF สามารถคำนวณได้ด้วยสมการ ดังสมการ 2

$$IDF(\text{ของคำ ๆ หนึ่ง}) = log(\frac{\text{จำนวนเอกสารทั้งหมดที่ใช้พิจารณา}}{\text{จำนวนเอกสารที่มีคำ ๆ นั้นปรากฏอยู่}}) \qquad (2)$$

เมื่อนำการคำนวณทั้งสองส่วนมารวมกัน เราจะได้การคำนวณ TF-IDF ดังต่อไปนี้ ดังสมการ 3

$$TFIDF = TF\ x\ IDF \quad (3)$$

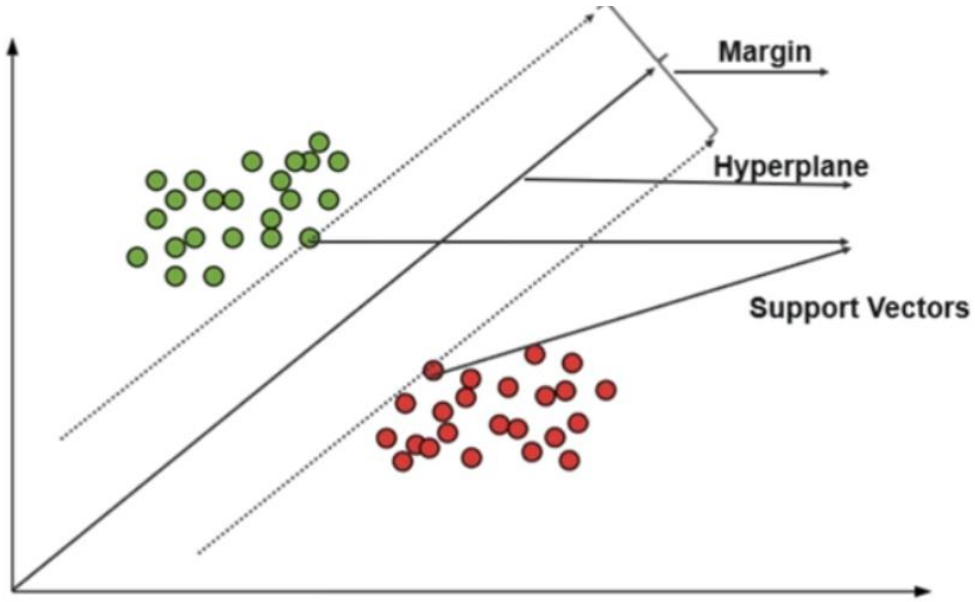
แบ่งชุดข้อมูลออกเป็นชุดข้อมูลสำหรับการฝึกแบบจำลอง (training dataset) 80% และชุดข้อมูลสำหรับการทดสอบแบบจำลอง (testing dataset) 20% random_state เท่ากับ 42 และสร้างแบบจำลองเพื่อประเมินประสิทธิภาพความถูกต้องและความแม่นยำ 8 แบบ ได้แก่

1. Support Vector Classification (SVC)
2. Logistic Regression
3. Random Forest
4. K-Nearest Neighbors
5. Gradient Boosting
6. AdaBoost Classifier
7. Gaussian Naïve Bayes
8. Decision Tree

อีกทั้งยังใช้เทคนิค OneVsRestClassifier เป็นคลาสย่อยของ Classifier ในไลบรารี scikit-learn ที่ใช้เพื่อจำแนกประเภทแบบ multi-class โดยสร้างแบบจำลองย่อยแยกกันสำหรับแต่ละคลาส เช่น ในงานวิจัยนี้มีชุดข้อมูลทั้งหมด 25 คลาส ซึ่งแบบจำลอง OneVsRestClassifier จะสร้างแบบจำลองย่อย 25 แบบจำลอง แต่ละแบบจำลองจะจำแนกประเภทตัวอย่างออกเป็นคลาสเดียว ซึ่งรายละเอียดของแบบจำลองที่ใช้ในงานวิจัยมีดังนี้

3.4.1 Support Vector Classification (SVC) (ชิตพงษ์ กิตตินราดร, 2563) (Premanand, 2566) (Mr.P L, 2561)

เป็นโมเดลจำแนกประเภทแบบ supervised learning ที่ใช้หลักการของ Hyperplane ในการหาเส้นแบ่งข้อมูลออกเป็นสองกลุ่มหรือมากกว่า ซึ่งทั้งยืดหยุ่นและทำงานได้ดีโดยเฉพาะอย่างยิ่งเมื่อข้อมูลมีความซับซ้อน (หลาย Feature) แต่จำนวนตัวอย่างไม่มาก หลายคนมักสับสนกับคำว่า SVM และ SVC คำตอบง่าย ๆ ก็คือ ถ้าไฮเปอร์เพลนที่เราใช้ในการจำแนกประเภทนั้น มีเงื่อนไขเชิงเส้น จะเรียก SVC



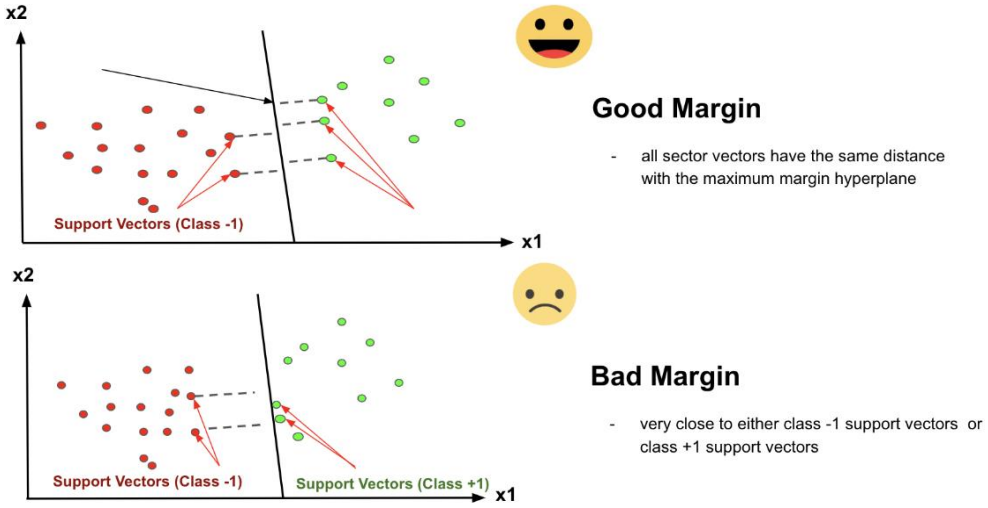
ภาพประกอบ 74 *Support Vector Machine*

ที่มา : (Premanand, 2566)

ระยะห่างของเวกเตอร์จากไฮเปอร์เพลนเรียกว่าระยะขอบ (margin) ซึ่งเป็นการแยกเส้นไปยังจุดคลาสที่ใกล้ที่สุด เราต้องการเลือกไฮเปอร์เพลนที่เพิ่มระยะขอบระหว่างคลาสให้สูงสุด กราฟด้านล่างแสดงระยะขอบที่ดีและระยะขอบที่ไม่ดี ซึ่ง Margin นี้แบ่งออกเป็น 2 ประเภท คือ

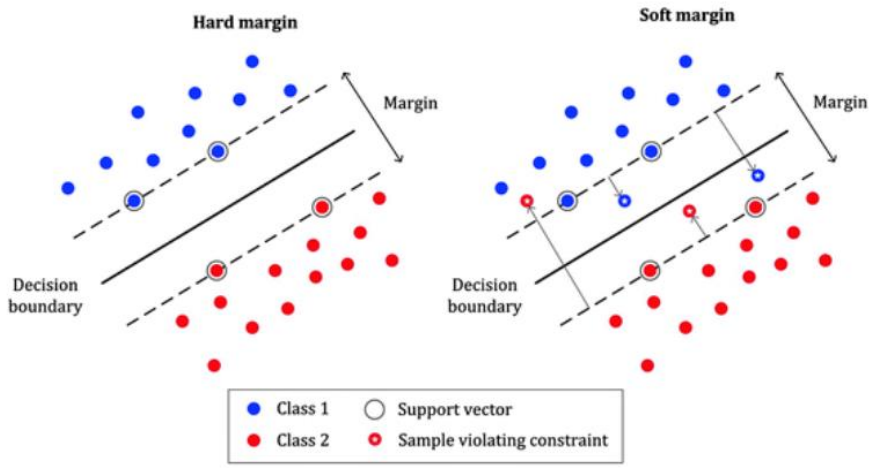
- Soft Margin เนื่องจากข้อมูลในโลกแห่งความเป็นจริงส่วนใหญ่ไม่สามารถแยกออกเป็นเส้นตรงได้อย่างสมบูรณ์ จึงทำให้เส้นแบ่งนั้นมีความกว้างมากที่สุดเท่าที่จะทำได้และยอมให้มีข้อมูลบางข้อมูลอยู่ระหว่างเส้นแบ่งของเราบ้าง เพื่อไม่ให้เส้นมันเพี้ยนมากจนเกินไป (Mr.P L, 2561)

- Hard Margin คือเส้นแบ่งแคบเกินไปและไม่ยอมให้มีข้อมูลอยู่ระหว่างเส้นแบ่งและถ้ามีข้อมูลที่เป็น outlier เกิดขึ้น ดังภาพประกอบ 74-76



ภาพประกอบ 75 แสดงเส้น *Good margin* และ *Bad margin*

ที่มา : (Premanand, 2566)



ภาพประกอบ 76 แสดง *Hard margin* และ *Soft margin*

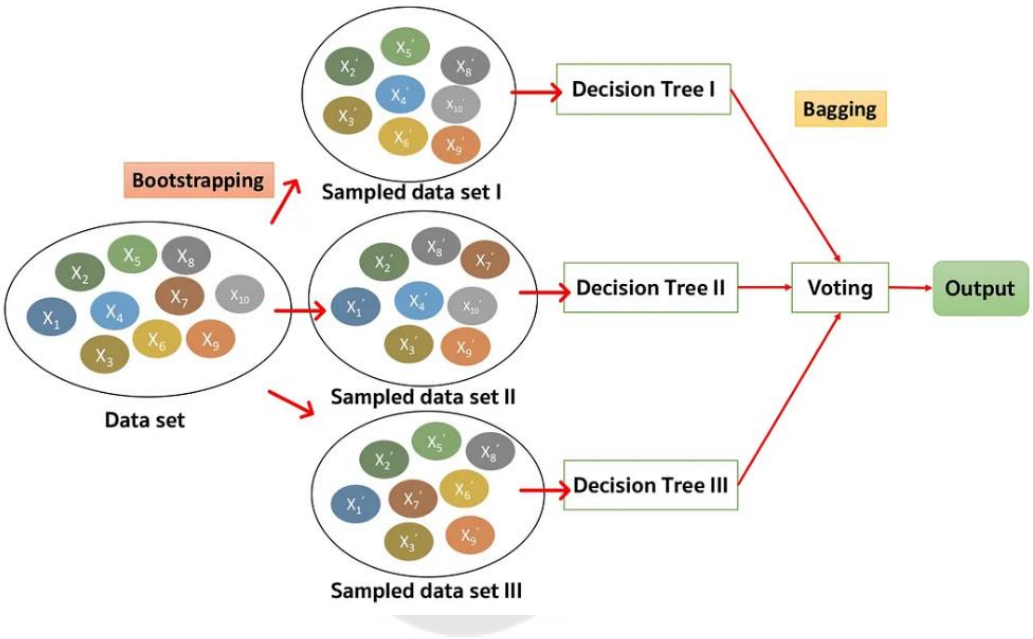
3.4.2 Logistic Regression

เป็นโมเดลการเรียนรู้ของเครื่อง (Machine Learning) ประเภทการจำแนกประเภท (Classification) ที่ใช้สำหรับจำแนกข้อมูลออกเป็นสองคลาส โดยใช้สมการเชิงเส้นในการประมาณความน่าจะเป็นที่ข้อมูลจะอยู่ในคลาสใดคลาสหนึ่ง จะอยู่ระหว่าง

0 ถึง 1 ข้อดีคือ เข้าใจง่าย คำนวณได้รวดเร็ว สามารถประยุกต์ใช้กับปัญหาการจำแนกประเภทได้หลากหลาย

3.4.3 Random Forest

คือสร้างแบบจำลองจาก Decision Tree หลาย ๆ แบบจำลองย่อย ๆ โดยแต่ละแบบจำลองจะได้รับ data set ไม่เหมือนกัน ซึ่งเป็น subset ของ data set ทั้งหมด ตอนทำ prediction ก็ให้แต่ละ Decision Tree ทำ prediction ของใครของมัน และคำนวณผล prediction ด้วยการ vote output ที่ ถูกเลือกโดย Decision Tree มากที่สุด (กรณี classification) ซึ่ง Decision Tree แต่ละแบบจำลองใน Random Forest ถือว่าเป็น weak learner กล่าวคือเป็นแบบจำลองที่ไม่เก่งเท่าไหร่ แต่พอนำเอาแต่ละ Decision Tree มาทำ prediction ร่วมกัน ก็จะได้แบบจำลองรวมที่มีความเก่ง และแม่นยำมากกว่า Decision Tree ที่ทำ prediction แบบเดี่ยว ๆ (Daroontham, 2561a) ดังภาพประกอบ 77



ภาพประกอบ 77 หลักการทำ *Random Forest*

ที่มา : (Daroontham, 2561a)

3.4.4 K-Nearest Neighbors

หรือเรียกย่อ ๆ ว่า KNN ใช้สำหรับจำแนกข้อมูลออกเป็นสองคลาสหรือมากกว่า โดยใช้วิธีการโหวตจากเพื่อนบ้านที่ใกล้ที่สุด ข้อมูลเพื่อนบ้านที่ใกล้ที่สุดของข้อมูลใหม่จะถูกโหวตเพื่อกำหนดคลาสของข้อมูลใหม่ หากเพื่อนบ้านที่ใกล้ที่สุด K ตัว ของข้อมูลใหม่มีคลาสเดียวกัน

โมเดล KNN จะจำแนกข้อมูลใหม่ให้เป็นคลาสนั้น หากเพื่อนบ้านที่ใกล้ที่สุด K ตัว ของข้อมูลใหม่มีคลาสต่างกัน โมเดล KNN จะจำแนกข้อมูลใหม่ให้เป็นคลาสที่มีจำนวนเพื่อนบ้านมากที่สุด

3.4.5 Gradient Boosting

เป็นโมเดลการเรียนรู้ของเครื่องประเภท Ensemble Learning ที่รวมโมเดลย่อย (Weak Learners) จำนวนมากเข้าด้วยกันเพื่อสร้างโมเดลที่มีประสิทธิภาพมากขึ้น

Gradient Boosting ทำงานโดยสร้างโมเดลย่อยขึ้นมาทีละโมเดล โดยโมเดลย่อยแต่ละโมเดลจะพยายามปรับปรุงความผิดพลาดของโมเดลก่อนหน้า

โมเดลย่อยแต่ละโมเดลจะสร้างขึ้นมาโดยใช้ Loss Function เพื่อวัดความผิดพลาดของโมเดลก่อนหน้า จากนั้นจะปรับน้ำหนักของโมเดลย่อยนั้นเพื่อให้ความผิดพลาดลดลง

3.4.6 AdaBoost classifier

หลักการทำงานจะเหมือนกัน Gradient Boosting แต่ต่างกันที่วิธีการปรับน้ำหนักตัวอย่างข้อมูล และ AdaBoost classifier จะปรับน้ำหนักตัวอย่างข้อมูลในชุดข้อมูลการฝึก (Training Set) เพื่อให้ความผิดพลาดของโมเดลย่อยลดลง โดยตัวอย่างข้อมูลที่โมเดลย่อยทำนายผิดพลาดจะมีน้ำหนักเพิ่มขึ้น ในขณะที่ตัวอย่างข้อมูลที่โมเดลย่อยทำนายถูกต้องจะมีน้ำหนักลดลง

3.4.7 Gaussian Naïve ayes

ใช้หลักการทำงานโดยอาศัยความน่าจะเป็นของข้อมูลแต่ละ feature ในการจำแนกข้อมูล

3.4.8 Decision Tree

ใช้แนวคิดการจำแนกข้อมูลแบบต้นไม้ (Tree-based Classification) โดยจะเริ่มต้นที่รากของต้นไม้ จากนั้นจะพิจารณา feature ของข้อมูลแต่ละ Training Set หากค่าของ feature นั้นเป็นไปตามเงื่อนไขที่กำหนด ก็จะเข้าสู่กิ่งก้านสาขานั้น ซึ่งกระบวนการนี้จะดำเนินต่อไปจนกว่าจะถึงใบไม้ของต้นไม้ ซึ่งใบไม้จะแสดงถึงคลาสของข้อมูลนั้น

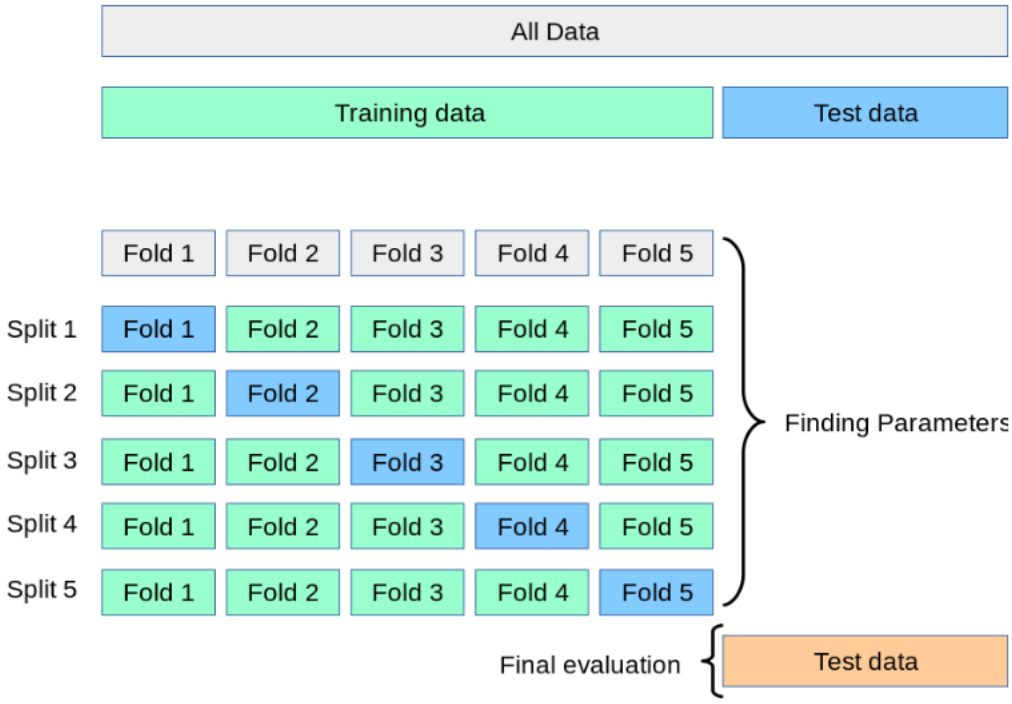
3.5 การทดสอบประสิทธิภาพของแบบจำลอง (Cross Validation Model)

กระบวนการทำงานของ Cross Validation แบบ K-fold จะแบ่งชุดข้อมูลออกเป็น K ชุด ชุดละเท่า ๆ กัน จากนั้นใช้ชุดย่อยหนึ่งชุดสำหรับการฝึกแบบจำลองและใช้ชุดย่อยที่เหลือสำหรับทดสอบแบบจำลองซ้ำ K ครั้ง ตัวอย่างเช่น

- ในการทดลองครั้งแรก ใช้ fold ที่ 1 สำหรับทดสอบแบบจำลอง ในขณะที่ใช้ fold ที่ 2, 3, 4, และ 5 สำหรับฝึกอบรมแบบจำลอง จากนั้นจึงคำนวณค่าความคลาดเคลื่อนของแบบจำลอง จากข้อมูลใน fold ที่ 1

- ในการทดลองครั้งที่สอง ใช้ fold ที่ 2 สำหรับทดสอบแบบจำลอง ในขณะที่ใช้ fold ที่ 1, 3, 4, และ 5 สำหรับฝึกอบรมแบบจำลอง จากนั้นจึงคำนวณค่าความคลาดเคลื่อนของแบบจำลอง จากข้อมูลใน fold ที่ 2

ทำซ้ำกระบวนการนี้ทั้งหมด 5 ครั้ง โดยสุ่มเลือกชุดย่อยสำหรับทดสอบในแต่ละครั้ง ค่าความคลาดเคลื่อนของแบบจำลองเฉลี่ยจาก 5 ครั้ง จะเป็นค่าความคลาดเคลื่อนของแบบจำลองที่แท้จริงนั่นเอง ดังภาพประกอบ 78



ภาพประกอบ 78 กระบวนการทำงานของ *Cross Validation* แบบ *K-fold*

ที่มา : (scikit-learn, 2023)

3.6 **การประเมินแบบจำลอง** (Evaluation)

ในการประเมินผลการทดลองการคัดกรองผู้สมัครจากประวัติย่อ เพื่อเปรียบเทียบประสิทธิภาพของแบบจำลองต่าง ๆ ที่ใช้ในการทดลอง ผู้วิจัยเลือกใช้ ค่า Accuracy, ค่า F1 score เป็นตัวชี้วัดการประเมินผลของแบบจำลอง อีกทั้งใช้ Cross Validation แบบ 5 fold เพื่อประเมินค่าความคลาดเคลื่อนของตัวชี้วัดต่าง ๆ เช่น accuracy, precision, recall, และ F1 score อีกด้วย

3.6.1 Accuracy อัตราส่วนของการทำนายที่ถูกต้องทั้งหมดต่อการทำนายทั้งหมด โดยคำนวณจากสมการ 4

$$\text{Accuracy} = (TP + TN) / (TP + TN + FP + FN) \quad (4)$$

3.6.2 Precision: อัตราส่วนของการทำนายที่ถูกต้องต่อการทำนายทั้งหมดที่เป็นคลาสนั้น ๆ โดยคำนวณจากสมการ 5

$$\text{Precision} = TP / (TP + FP) \quad (5)$$

3.6.3 Recall: อัตราส่วนของการทำนายที่ถูกต้องต่อข้อมูลจริงทั้งหมดที่เป็นคลาสนั้น ๆ โดยคำนวณจากสมการ 6

$$\text{Recall} = TP / (TP + FN) \quad (6)$$

3.6.4 F1 score: ค่าเฉลี่ยแบบ harmonic mean ระหว่าง precision และ recall โดยคำนวณจากสมการ 7

$$\text{F1 score} = 2 * (\text{precision} * \text{recall}) / (\text{precision} + \text{recall}) \quad (7)$$

3.7 **การหาคุณลักษณะที่สำคัญ** (Feature Importance)

ตัวแปรหรือคุณลักษณะที่สำคัญในชุดข้อมูลของงานวิจัยนี้คือคำต่าง ๆ ที่อยู่ในประวัติย่อ เพื่อทราบว่าคำใดที่ส่งผลต่อการจำแนกประเภทงาน โดยจะใช้วิธีการหาคุณลักษณะที่สำคัญ 2 วิธี ดังนี้

1. หาค่า Coefficients

- สร้างแบบจำลอง: ทำการสร้างโมเดล Logistic Regression โดยใช้ข้อมูลที่ถูกสกัดคุณลักษณะด้วย TF-IDF ใช้ในการแปลงข้อมูลข้อความเป็นเวกเตอร์ของคุณลักษณะ (Features) โดยคำนึงถึงความถี่ของคำแต่ละคำในเอกสาร และความสำคัญของคำนั้นในเอกสารทั้งหมด

- ฝึกแบบจำลอง: ทำการฝึกแบบจำลองที่ได้มาจากการสร้างแบบจำลองก่อนหน้าด้วยชุดข้อมูลการเรียนรู้ (Training data)

- คำนวณหาคุณลักษณะที่สำคัญ (Feature Importance): เมื่อแบบจำลองที่ถูกเรียนรู้แล้ว สามารถดึงค่า coefficients หรือค่าความสำคัญของคุณลักษณะ (importance) จากแบบจำลองได้ ซึ่งในกรณี Logistic Regression ค่า coefficients จะบ่งบอกถึงปริมาณของการเปลี่ยนแปลงในผลลัพธ์ (ค่าของตัวแปรตาม) เมื่อมีการเปลี่ยนแปลงในตัวแปรอิสระ (ค่าของตัวแปรอิสระ) โดยที่ค่า coefficients ที่มากที่สุดจะบ่งบอกถึงคุณลักษณะที่มีผลต่อการทำนายมากที่สุด

- แสดงผลลัพธ์: นำค่า coefficients หรือค่าความสำคัญของคุณลักษณะที่ได้มาแสดงผลเป็นกราฟของคุณลักษณะที่มีความสำคัญที่สุด เพื่อให้สามารถทำความเข้าใจถึงคุณลักษณะที่มีผลต่อผลลัพธ์มากที่สุดได้โดยง่ายและชัดเจน

2. หาค่า SHAP (SHapley Additive exPlanations)

SHAP เป็นวิธีการอธิบายแบบจำลอง Machine Learning ช่วยวิเคราะห์ว่าคุณลักษณะ (Feature) แต่ละตัวส่งผลต่อผลลัพธ์การทำนายของแบบจำลองอย่างไร และสามารถวิเคราะห์แบบจำลองการจำแนกประเภทสายงานจากประวัติย่อว่าคำใดที่มีผลต่อการจำแนกประเภท โดยเริ่มต้นจะคำนวณ SHAP Values ด้วยการใช้ไลบรารี shap.TreeExplainer โดยใช้แบบจำลองที่ถูกเรียนรู้แล้ว และข้อมูลทดสอบ (X_test) ซึ่งแบบจำลองและข้อมูลทดสอบมาจากกระบวนการสร้างแบบจำลองด้วยอัลกอริทึม Random Forest และทำการคำนวณค่า SHAP values สำหรับแต่ละคำ แยกตามแต่ประเภทงานทั้งหมด 25 ประเภทงาน เมื่อได้ค่า SHAP values จะทำการแสดงผลด้วยกราฟ โดยกราฟจะแสดงคำที่มีค่า SHAP Values สูงสุดที่มีส่วนสำคัญในการจำแนกประเภทงาน

บทที่ 4
การทดลอง และผลลัพธ์ของการวิจัย

หลังจากผ่านกระบวนการสร้างแบบจำลองเพื่อจำแนกประเภทงานจากประวัติย่อด้วยชุดข้อมูลสาธารณะจากเว็บไซต์ Kaggle โดยใช้เทคนิคการเรียนรู้ของเครื่องด้วยอัลกอริทึม Support Vector Classification (SVC), Logistic Regression, Random Forest, K-Nearest Neighbors (KNN), Gradient Boosting, AdaBoost Classifier, Gaussian Naïve Bayes, Decision Tree เรียบร้อยแล้ว ก็จะมาวัดประสิทธิภาพของแต่ละแบบจำลองจากค่าความถูกต้อง (Accuracy) ค่าความแม่นยำ (Precision) ค่าความไว (Recall ค่าเอฟวัน (F1-Score) ซึ่งผลลัพธ์ที่ได้จากการเปรียบเทียบการประเมินประสิทธิภาพโดยรวมในแต่ละอัลกอริทึม การวัดประสิทธิภาพของแบบจำลองในแต่ละอัลกอริทึมโดยแยกตามแต่ละประเภทงาน และการหาค่าคุณลักษณะที่สำคัญในการจำแนกประเภทงานจากประวัติย่อ แสดงรายละเอียดดังต่อไปนี้

4.1 ประสิทธิภาพของแบบจำลอง

เมื่อนำผลลัพธ์ของแบบจำลองในแต่ละอัลกอริทึมมาเปรียบเทียบประสิทธิภาพกันพบว่าแบบจำลอง SVC, Logistic Regression และ Random Forest มีค่าความถูกต้อง (Accuracy) มากที่สุดอยู่ที่ 99.48% เมื่อดูค่า Cross Validation ประกอบพบว่า SVC ได้ค่าสูงที่สุด และมีค่า Precision 99.5%, ค่า Recall 99.71%, ค่า F1-Score 99.58% ดังตาราง 17

ตาราง 17 ผลเปรียบเทียบการประเมินประสิทธิภาพโดยรวมในแต่ละอัลกอริทึม

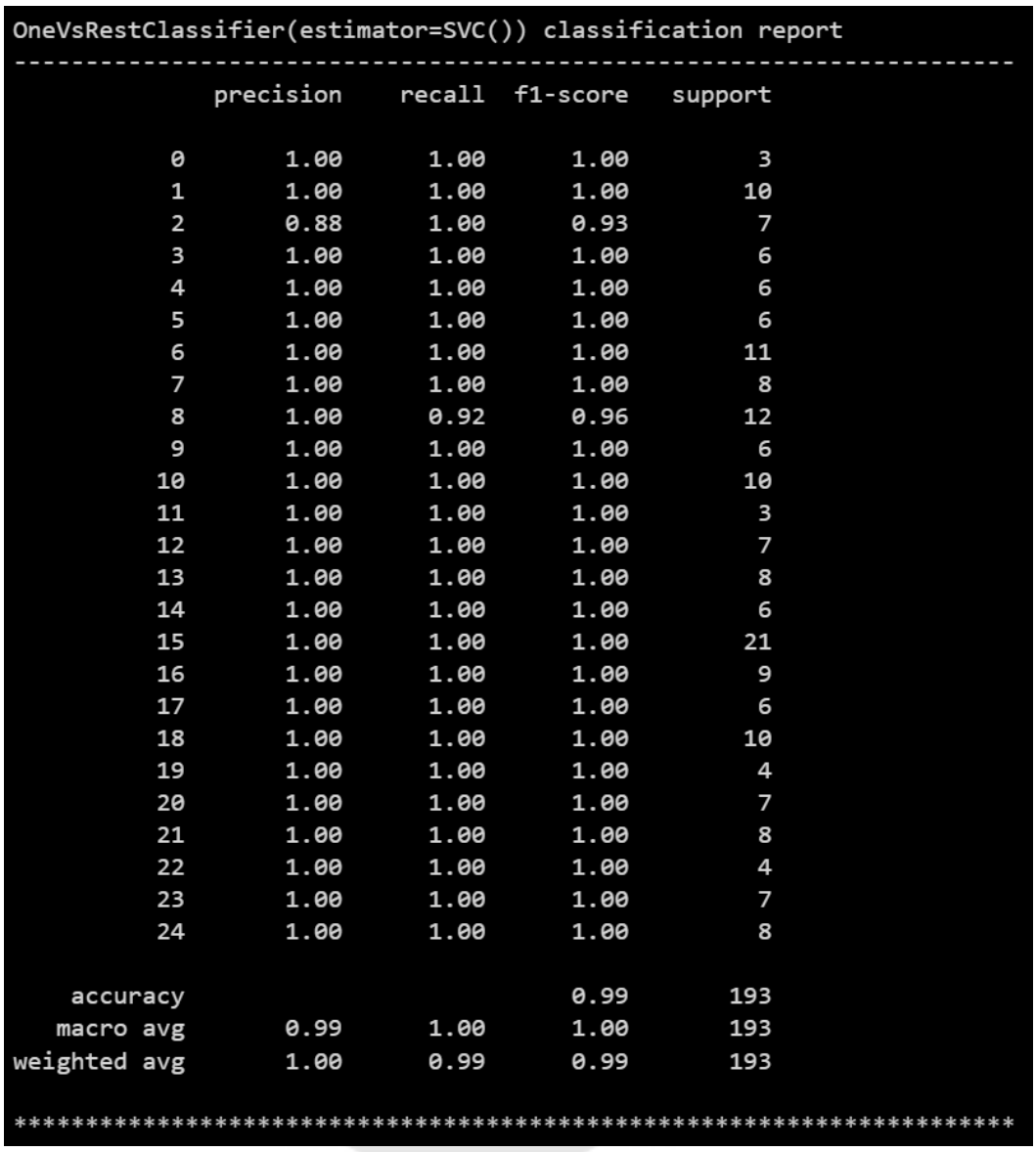
Model	Precision	Recall	F1-Score	Accuracy	Cross Validation (5-folds)
SVC	0.995	0.9971	0.9958	0.9948	0.995
Logistic Regression	0.9976	0.9971	0.9973	0.9948	0.992
Random Forest	0.9976	0.9971	0.9973	0.9948	0.991
Gradient Boosting	0.9892	0.9704	0.9727	0.9844	0.991
AdaBoost	0.995	0.9971	0.9958	0.9948	0.99
Decision Tree	0.9933	0.9971	0.9948	0.9948	0.987
Gaussian Naïve Bayes	0.9976	0.9971	0.9973	0.9948	0.984
KNN	0.9761	0.9761	0.9748	0.9792	0.971

ข้อมูลประกอบไปด้วย 25 คลาส ดังตาราง 18

ตาราง 18 แสดงชื่อคลาสและชื่อประเภทงาน

Class	Category Name
0	Advocate
1	Arts
2	Automation Testing
3	Blockchain
4	Business Analyst
5	Civil Engineer
6	Data Science
7	Database
8	DevOps Engineer
9	DotNet Developer
10	ETL Developer
11	Electrical Engineering
12	HR
13	Hadoop
14	Health and fitness
15	Java Developer
16	Mechanical Engineer
17	Network Security Engineer
18	Operations Manager
19	PMO
20	Python Developer
21	SAP Developer
22	Sales
23	Testing
24	Web Designing

เมื่อทำการวัดประสิทธิภาพของแต่ละแบบจำลองโดยแยกตามประเภทงานทั้ง 25 ประเภทงาน พบว่า class ส่วนใหญ่ทำนายได้ถูกต้องแม่นยำ ดังภาพประกอบ 79-86

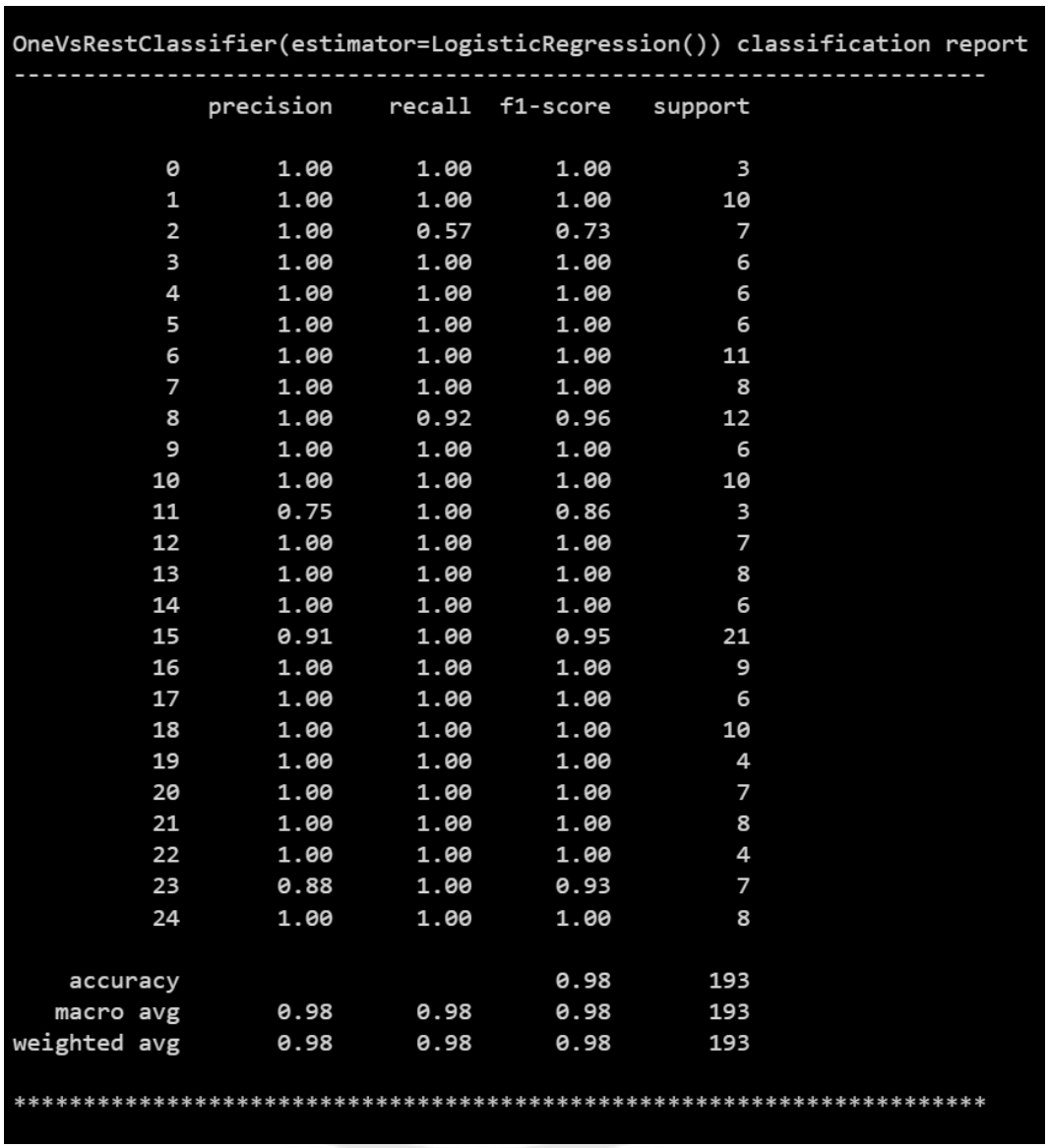


OneVsRestClassifier(estimator=SVC()) classification report

	precision	recall	f1-score	support
0	1.00	1.00	1.00	3
1	1.00	1.00	1.00	10
2	0.88	1.00	0.93	7
3	1.00	1.00	1.00	6
4	1.00	1.00	1.00	6
5	1.00	1.00	1.00	6
6	1.00	1.00	1.00	11
7	1.00	1.00	1.00	8
8	1.00	0.92	0.96	12
9	1.00	1.00	1.00	6
10	1.00	1.00	1.00	10
11	1.00	1.00	1.00	3
12	1.00	1.00	1.00	7
13	1.00	1.00	1.00	8
14	1.00	1.00	1.00	6
15	1.00	1.00	1.00	21
16	1.00	1.00	1.00	9
17	1.00	1.00	1.00	6
18	1.00	1.00	1.00	10
19	1.00	1.00	1.00	4
20	1.00	1.00	1.00	7
21	1.00	1.00	1.00	8
22	1.00	1.00	1.00	4
23	1.00	1.00	1.00	7
24	1.00	1.00	1.00	8
accuracy			0.99	193
macro avg	0.99	1.00	1.00	193
weighted avg	1.00	0.99	0.99	193

****

ภาพประกอบ 79 แสดงค่า *precision, recall, f1 score* ของแต่ละคลาสของ *SVC Model*



OneVsRestClassifier(estimator=LogisticRegression()) classification report

	precision	recall	f1-score	support
0	1.00	1.00	1.00	3
1	1.00	1.00	1.00	10
2	1.00	0.57	0.73	7
3	1.00	1.00	1.00	6
4	1.00	1.00	1.00	6
5	1.00	1.00	1.00	6
6	1.00	1.00	1.00	11
7	1.00	1.00	1.00	8
8	1.00	0.92	0.96	12
9	1.00	1.00	1.00	6
10	1.00	1.00	1.00	10
11	0.75	1.00	0.86	3
12	1.00	1.00	1.00	7
13	1.00	1.00	1.00	8
14	1.00	1.00	1.00	6
15	0.91	1.00	0.95	21
16	1.00	1.00	1.00	9
17	1.00	1.00	1.00	6
18	1.00	1.00	1.00	10
19	1.00	1.00	1.00	4
20	1.00	1.00	1.00	7
21	1.00	1.00	1.00	8
22	1.00	1.00	1.00	4
23	0.88	1.00	0.93	7
24	1.00	1.00	1.00	8
accuracy			0.98	193
macro avg	0.98	0.98	0.98	193
weighted avg	0.98	0.98	0.98	193

****

ภาพประกอบ 80 แสดงค่า *precision, recall, f1 score* ในแต่ละคลาสของ *Logistic Regression Model*

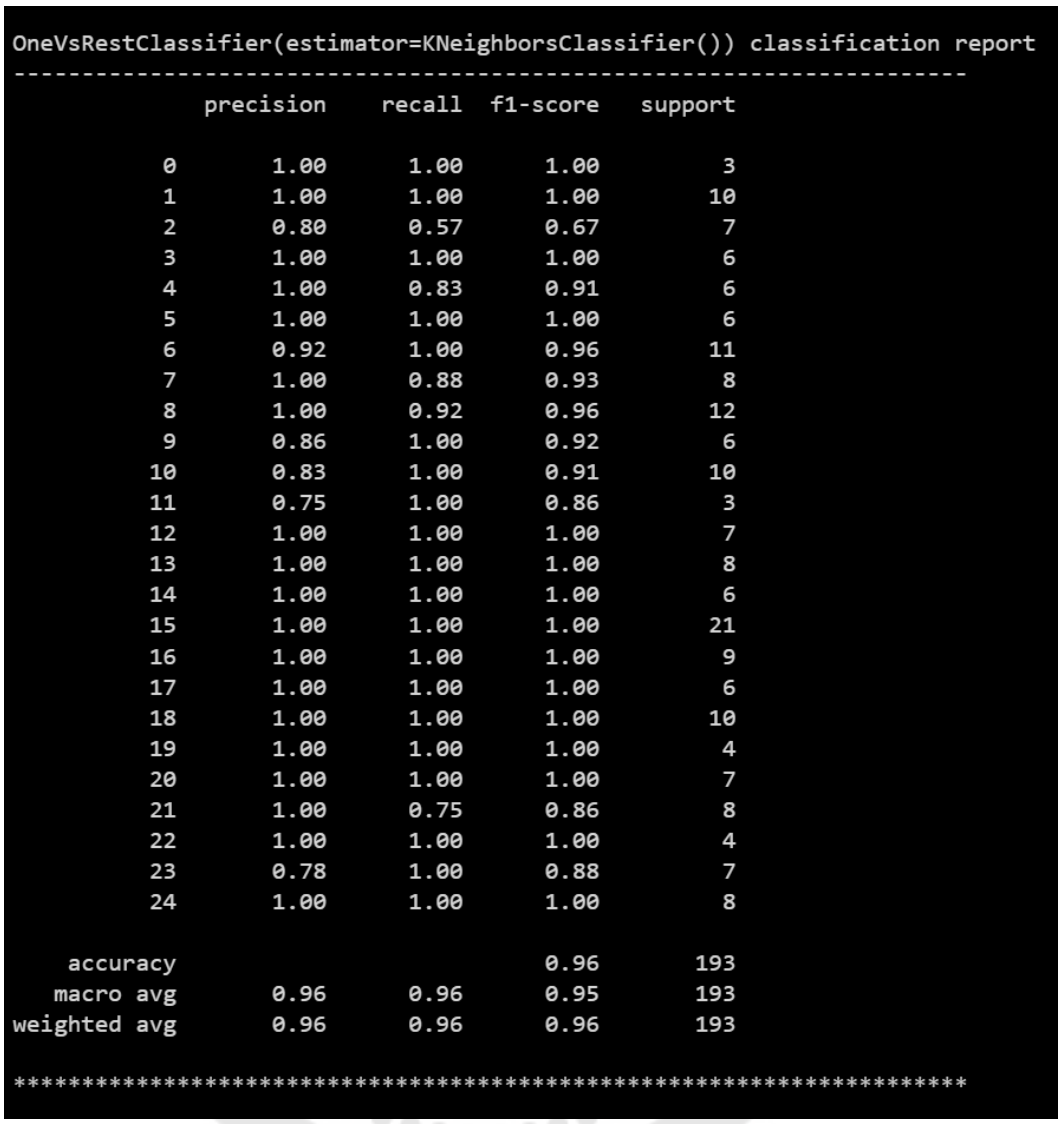
```
OneVsRestClassifier(estimator=RandomForestClassifier()) classification report
----------------------------------------------------------------------
              precision    recall  f1-score   support

           0       1.00      1.00      1.00         3
           1       1.00      1.00      1.00        10
           2       0.83      0.71      0.77         7
           3       1.00      1.00      1.00         6
           4       1.00      1.00      1.00         6
           5       1.00      1.00      1.00         6
           6       1.00      1.00      1.00        11
           7       1.00      1.00      1.00         8
           8       1.00      0.92      0.96        12
           9       1.00      1.00      1.00         6
          10       1.00      1.00      1.00        10
          11       1.00      1.00      1.00         3
          12       1.00      1.00      1.00         7
          13       1.00      1.00      1.00         8
          14       1.00      1.00      1.00         6
          15       0.91      1.00      0.95        21
          16       1.00      1.00      1.00         9
          17       1.00      1.00      1.00         6
          18       1.00      1.00      1.00        10
          19       1.00      1.00      1.00         4
          20       1.00      1.00      1.00         7
          21       1.00      1.00      1.00         8
          22       1.00      1.00      1.00         4
          23       1.00      1.00      1.00         7
          24       1.00      1.00      1.00         8

    accuracy                           0.98       193
   macro avg       0.99      0.99      0.99       193
weighted avg       0.98      0.98      0.98       193

**********************************************************************
```

ภาพประกอบ 81 แสดงค่า *precision, recall, f1 score* ในแต่ละคลาสของ *Random Forest Model*



OneVsRestClassifier(estimator=KNeighborsClassifier()) classification report

	precision	recall	f1-score	support
0	1.00	1.00	1.00	3
1	1.00	1.00	1.00	10
2	0.80	0.57	0.67	7
3	1.00	1.00	1.00	6
4	1.00	0.83	0.91	6
5	1.00	1.00	1.00	6
6	0.92	1.00	0.96	11
7	1.00	0.88	0.93	8
8	1.00	0.92	0.96	12
9	0.86	1.00	0.92	6
10	0.83	1.00	0.91	10
11	0.75	1.00	0.86	3
12	1.00	1.00	1.00	7
13	1.00	1.00	1.00	8
14	1.00	1.00	1.00	6
15	1.00	1.00	1.00	21
16	1.00	1.00	1.00	9
17	1.00	1.00	1.00	6
18	1.00	1.00	1.00	10
19	1.00	1.00	1.00	4
20	1.00	1.00	1.00	7
21	1.00	0.75	0.86	8
22	1.00	1.00	1.00	4
23	0.78	1.00	0.88	7
24	1.00	1.00	1.00	8
accuracy			0.96	193
macro avg	0.96	0.96	0.95	193
weighted avg	0.96	0.96	0.96	193

****

ภาพประกอบ 82 แสดงค่า *precision, recall, f1 score* ในแต่ละคลาสของ *KNN Model*

```
OneVsRestClassifier(estimator=GradientBoostingClassifier()) classification report
----------------------------------------------------------------------
              precision    recall  f1-score   support

           0       1.00      1.00      1.00         3
           1       1.00      1.00      1.00        10
           2       1.00      0.71      0.83         7
           3       1.00      1.00      1.00         6
           4       1.00      1.00      1.00         6
           5       1.00      1.00      1.00         6
           6       1.00      1.00      1.00        11
           7       1.00      1.00      1.00         8
           8       1.00      0.92      0.96        12
           9       1.00      1.00      1.00         6
          10       1.00      1.00      1.00        10
          11       1.00      1.00      1.00         3
          12       1.00      1.00      1.00         7
          13       1.00      1.00      1.00         8
          14       1.00      1.00      1.00         6
          15       1.00      1.00      1.00        21
          16       1.00      1.00      1.00         9
          17       1.00      1.00      1.00         6
          18       1.00      1.00      1.00        10
          19       1.00      1.00      1.00         4
          20       1.00      1.00      1.00         7
          21       1.00      1.00      1.00         8
          22       0.80      1.00      0.89         4
          23       0.78      1.00      0.88         7
          24       1.00      1.00      1.00         8

    accuracy                           0.98       193
   macro avg       0.98      0.99      0.98       193
weighted avg       0.99      0.98      0.98       193

**********************************************************************
```

ภาพประกอบ 83 แสดงค่า *precision, recall, f1 score* ในแต่ละคลาสของ *Gradient Boosting Model*

```
OneVsRestClassifier(estimator=AdaBoostClassifier()) classification report
---------------------------------------------------------------------
              precision    recall  f1-score   support

           0       1.00      1.00      1.00         3
           1       1.00      1.00      1.00        10
           2       1.00      1.00      1.00         7
           3       1.00      1.00      1.00         6
           4       1.00      1.00      1.00         6
           5       1.00      1.00      1.00         6
           6       1.00      1.00      1.00        11
           7       1.00      1.00      1.00         8
           8       1.00      0.92      0.96        12
           9       1.00      1.00      1.00         6
          10       1.00      1.00      1.00        10
          11       1.00      1.00      1.00         3
          12       1.00      1.00      1.00         7
          13       1.00      1.00      1.00         8
          14       1.00      1.00      1.00         6
          15       1.00      1.00      1.00        21
          16       1.00      1.00      1.00         9
          17       1.00      1.00      1.00         6
          18       1.00      1.00      1.00        10
          19       1.00      1.00      1.00         4
          20       1.00      1.00      1.00         7
          21       1.00      1.00      1.00         8
          22       0.80      1.00      0.89         4
          23       1.00      1.00      1.00         7
          24       1.00      1.00      1.00         8

    accuracy                           0.99       193
   macro avg       0.99      1.00      0.99       193
weighted avg       1.00      0.99      0.99       193

*********************************************************************
```

ภาพประกอบ 84 แสดงค่า *precision, recall, f1 score* ในแต่ละคลาสของ *Ada Boost Model*

OneVsRestClassifier(estimator=GaussianNB()) classification report

	precision	recall	f1-score	support
0	1.00	1.00	1.00	3
1	1.00	1.00	1.00	10
2	1.00	0.71	0.83	7
3	1.00	1.00	1.00	6
4	1.00	1.00	1.00	6
5	1.00	1.00	1.00	6
6	1.00	1.00	1.00	11
7	1.00	1.00	1.00	8
8	1.00	0.92	0.96	12
9	1.00	1.00	1.00	6
10	1.00	1.00	1.00	10
11	1.00	1.00	1.00	3
12	1.00	1.00	1.00	7
13	1.00	1.00	1.00	8
14	1.00	1.00	1.00	6
15	1.00	1.00	1.00	21
16	1.00	1.00	1.00	9
17	1.00	1.00	1.00	6
18	1.00	1.00	1.00	10
19	1.00	1.00	1.00	4
20	1.00	1.00	1.00	7
21	1.00	1.00	1.00	8
22	1.00	1.00	1.00	4
23	1.00	1.00	1.00	7
24	0.73	1.00	0.84	8
accuracy			0.98	193
macro avg	0.99	0.99	0.99	193
weighted avg	0.99	0.98	0.98	193

****

ภาพประกอบ 85 แสดงค่า *precision, recall, f1 score* ในแต่ละคลาสของ *Gaussian Naïve Bayes Model*

```
OneVsRestClassifier(estimator=DecisionTreeClassifier()) classification report
---------------------------------------------------------------------
              precision    recall  f1-score   support

           0       1.00      1.00      1.00         3
           1       1.00      1.00      1.00        10
           2       1.00      0.71      0.83         7
           3       1.00      1.00      1.00         6
           4       1.00      1.00      1.00         6
           5       1.00      1.00      1.00         6
           6       1.00      1.00      1.00        11
           7       1.00      1.00      1.00         8
           8       1.00      0.92      0.96        12
           9       1.00      1.00      1.00         6
          10       1.00      1.00      1.00        10
          11       1.00      1.00      1.00         3
          12       1.00      1.00      1.00         7
          13       1.00      1.00      1.00         8
          14       1.00      1.00      1.00         6
          15       1.00      1.00      1.00        21
          16       1.00      1.00      1.00         9
          17       1.00      1.00      1.00         6
          18       1.00      1.00      1.00        10
          19       1.00      1.00      1.00         4
          20       1.00      1.00      1.00         7
          21       1.00      1.00      1.00         8
          22       0.80      1.00      0.89         4
          23       0.78      1.00      0.88         7
          24       1.00      1.00      1.00         8

    accuracy                           0.98       193
   macro avg       0.98      0.99      0.98       193
weighted avg       0.99      0.98      0.98       193

*********************************************************************
```

ภาพประกอบ 86 แสดงค่า *precision, recall, f1 score* ในแต่ละคลาสของ *Decision Tree*

4.2 ประสิทธิภาพของคุณลักษณะที่สำคัญต่อการจำแนกประเภท (Feature Importance)

คุณลักษณะที่สำคัญในชุดข้อมูลของงานวิจัยนี้คือคำ (word) ที่อยู่ในประวัติย่อ เพื่อทราบว่าคำใดที่ส่งผลต่อการจำแนกประเภทงาน โดยจะใช้วิธีการหาคุณลักษณะที่สำคัญ 2 วิธี ได้แก่ การหาค่า Coefficients โดยใช้ข้อมูลที่ถูกสกัดคุณลักษณะด้วย TF-IDF ทำการสร้างแบบจำลอง Logistic Regression ฝึกฝนข้อมูลด้วยชุดข้อมูลฝึก และนำมาหาค่า Coefficients แสดงผลเป็นกราฟของคุณลักษณะที่มีความสำคัญที่สุดในแต่ละประเภทงานเรียงลำดับจากมากที่สุดไปน้อยที่สุด และการหาค่า SHAP โดยคำนวณ SHAP Values ด้วยการใช้ไลบรารี shap.TreeExplainer โดยใช้แบบจำลองที่ถูกเรียนรู้แล้ว และข้อมูลทดสอบ (X_test) ซึ่งแบบจำลองและข้อมูลทดสอบมาจากกระบวนการสร้างแบบจำลองด้วยอัลกอริทึม Random Forest และทำ

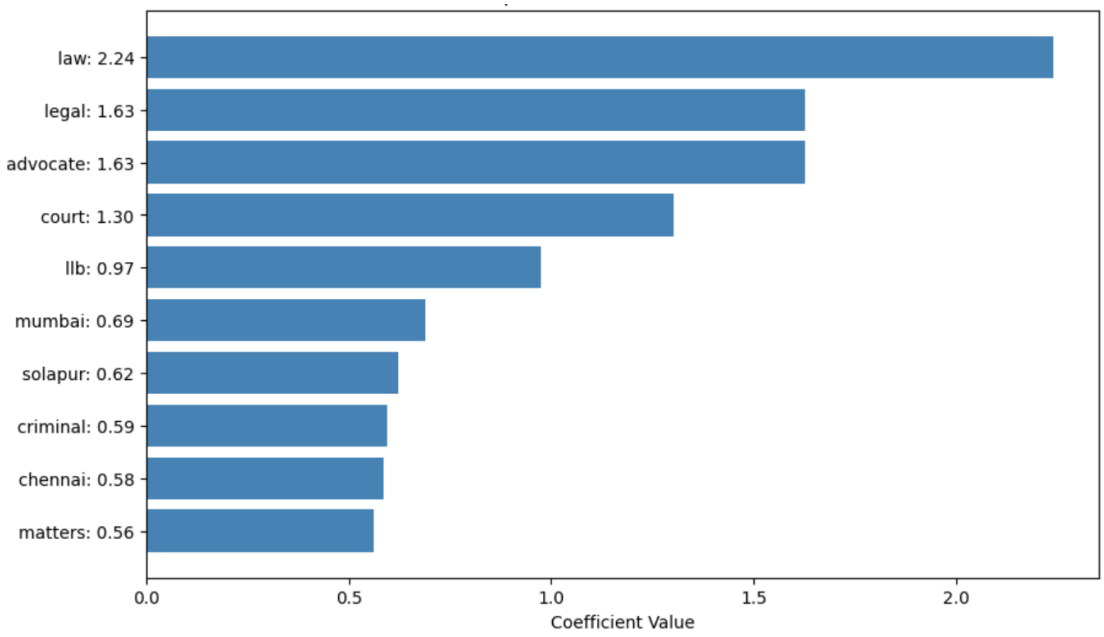
การคำนวณค่า SHAP values สำหรับแต่ละคำ แยกตามแต่ประเภทงานทั้งหมด 25 ประเภทงาน เมื่อได้ค่า SHAP values จะทำการแสดงผลด้วยกราฟ โดยกราฟจะแสดงคำที่มีค่า SHAP Values สูงสุดที่มีส่วนสำคัญในการจำแนกประเภทงาน

เมื่อเทียบผลลัพธ์ของหาคุณลักษณะที่สำคัญระหว่างแบบหาค่า Coefficients กับ SHAP Values พบว่าได้คำ (word) ที่มีความสำคัญต่อการจำแนกประเภทงานที่แตกต่างกัน เนื่องจากการทำงานของการหาค่า Coefficients และการหาค่า SHAP Values มีลักษณะการทำงานที่แตกต่างกัน กล่าวคือ

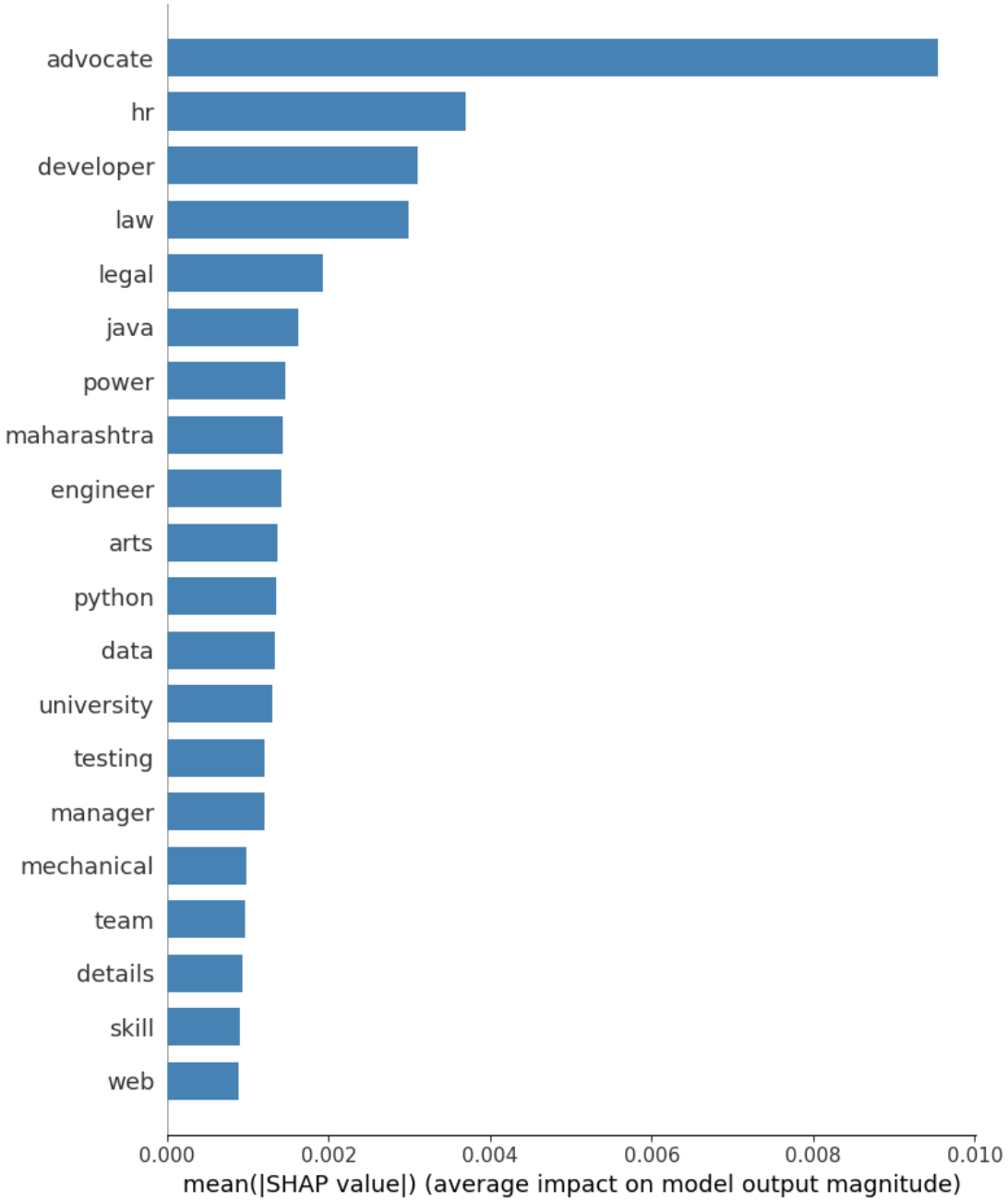
- การหาค่า Coefficients: ในแบบจำลองเชิงเส้น เช่น Logistic Regression การหาค่า coefficients จะใช้เพื่อระบุความสัมพันธ์ระหว่างตัวแปรอิสระกับตัวแปรตามโดยตรง โดย coefficient ที่มีค่ามากก็แสดงถึงความสำคัญของตัวแปรนั้น ๆ ในการทำนายผลลัพธ์ เช่น ถ้าโมเดล Linear Regression มี coefficient ของตัวแปร X_1 เป็นบวกและมีค่ามาก แสดงว่าเมื่อค่า X_1 เพิ่มขึ้น ผลลัพธ์จะมีแนวโน้มที่จะเพิ่มขึ้นด้วย

- การหาค่า SHAP Values: เป็นวิธีการที่ใช้ในการอธิบายการทำนายของโมเดลอย่างละเอียดและสอดคล้องกับความสำคัญของแต่ละตัวแปรอิสระ โดยการนำเสนอผลลัพธ์ที่แตกต่างจากค่าโดยเฉลี่ยของโมเดล (โดยใช้ตัวแปรทุกตัว) และทำการวิเคราะห์ว่าการเปลี่ยนแปลงในแต่ละตัวแปรจะส่งผลต่อการทำนายอย่างไร ซึ่งค่า SHAP values จะบ่งบอกถึงว่าแต่ละตัวแปรอิสระมีอิทธิพลต่อการทำนายอย่างไร โดยการบวก SHAP values ของทุกตัวแปรอิสระ และ SHAP base value (ค่าที่ใช้เป็น reference) จะได้ผลลัพธ์ที่ใกล้เคียงกับการทำนายของโมเดล

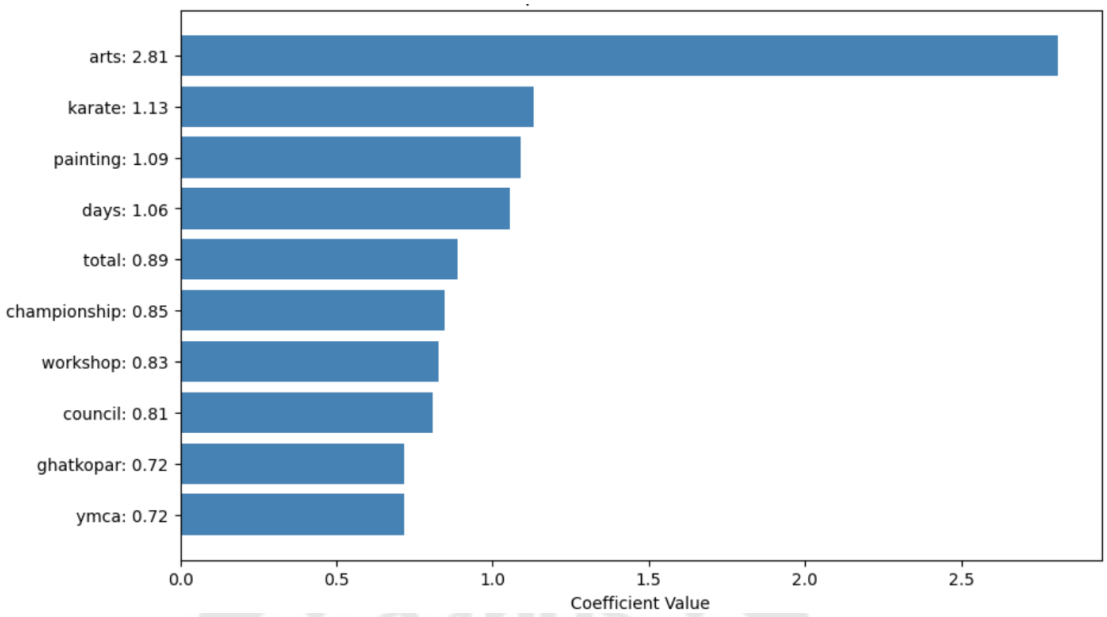
แสดงผลลัพธ์ของหาคุณลักษณะที่สำคัญระหว่างแบบหาค่า Coefficients กับ SHAP Values ในแต่ละประเภทงาน ดังภาพประกอบ 87-135



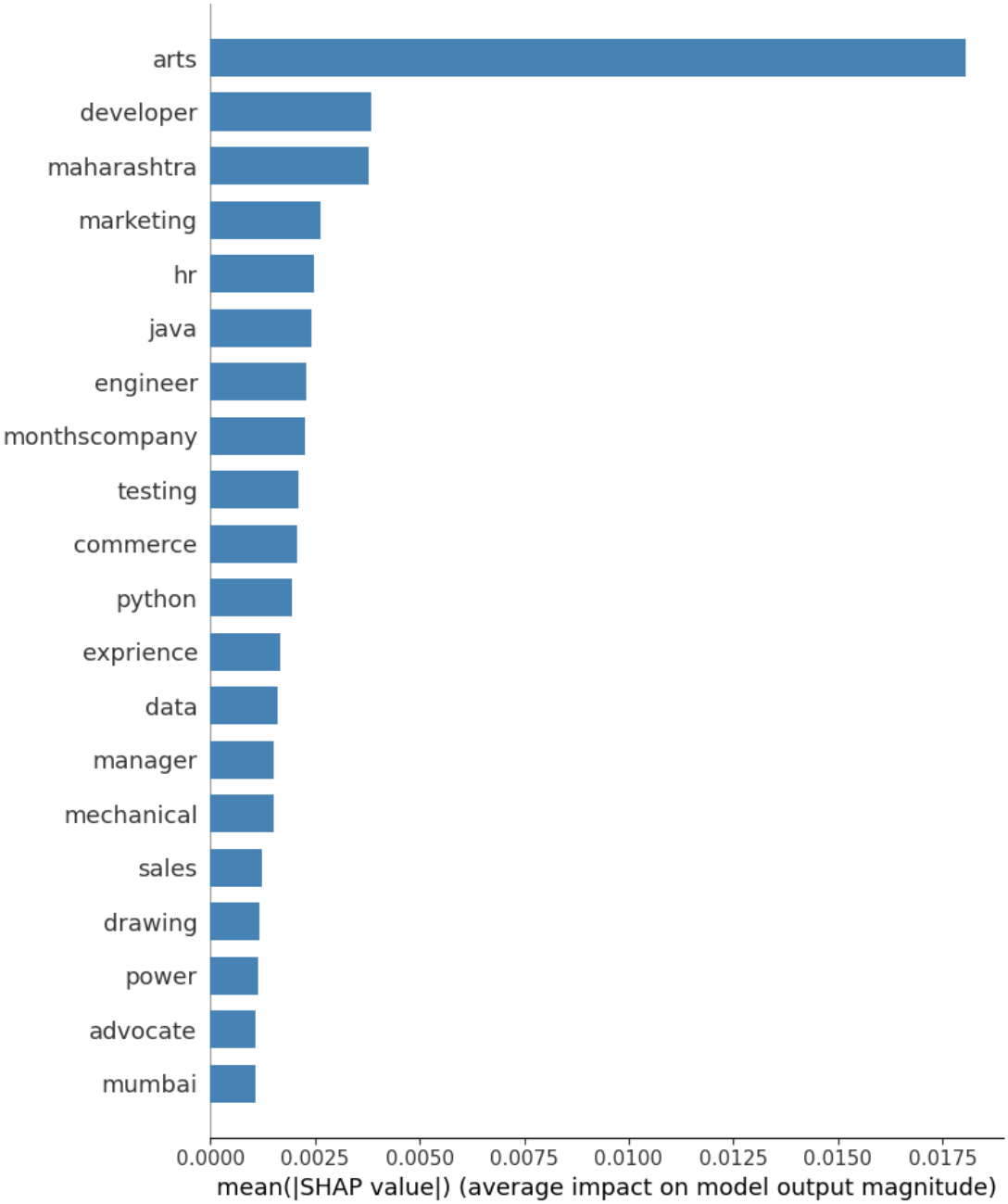
ภาพประกอบ 87แสดงค่า *Feature Importance* ของ *Class 0 – Advocate*



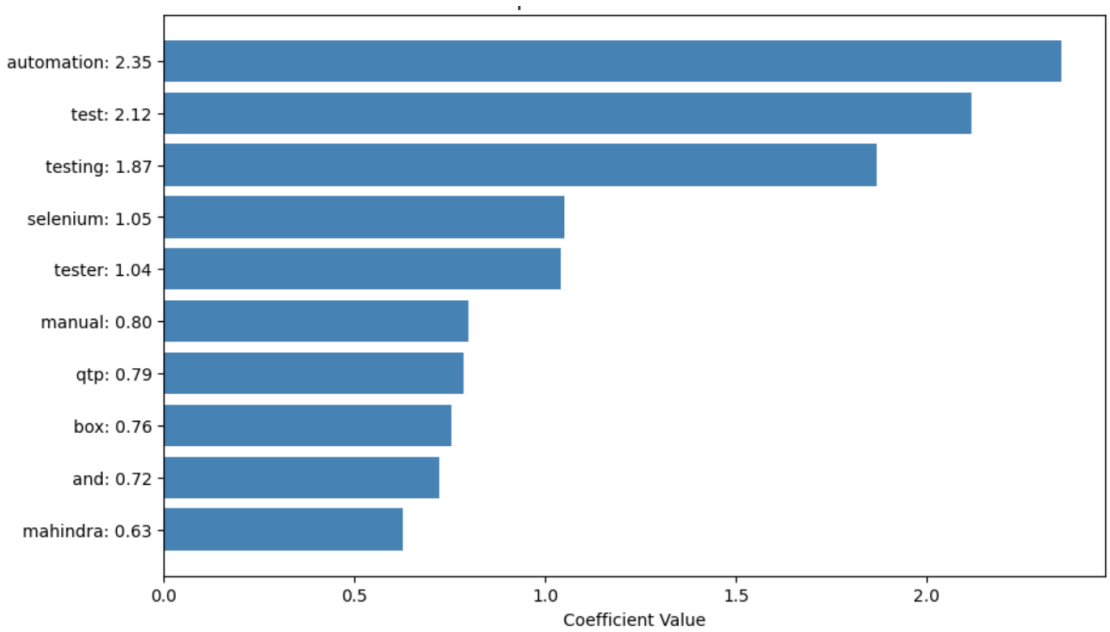
ภาพประกอบ 88 แสดงค่า *SHAP* ของ *Class 0 – Advocate*



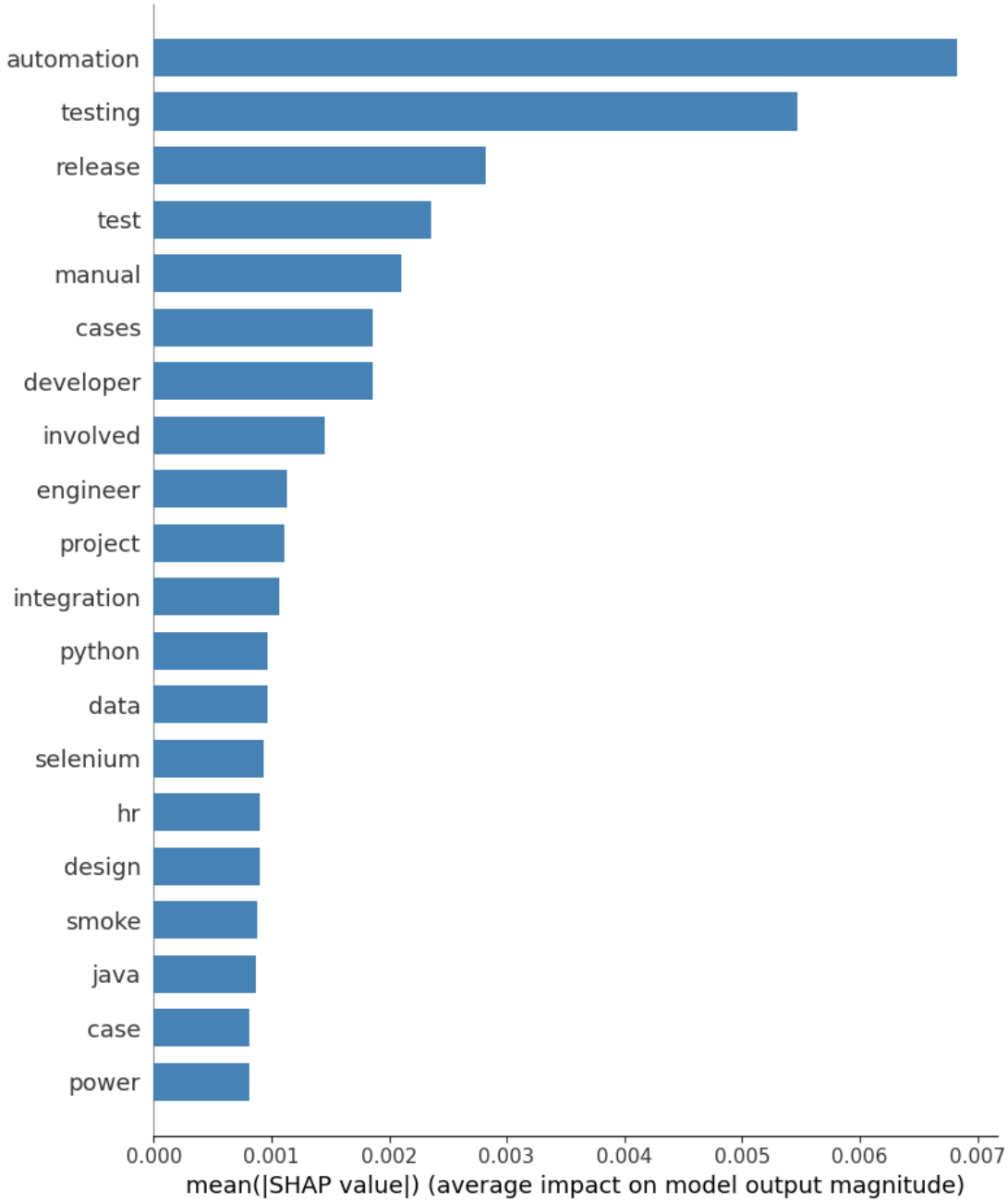
ภาพประกอบ 114 แสดงค่า Feature Importance ของ Class 1 – Arts



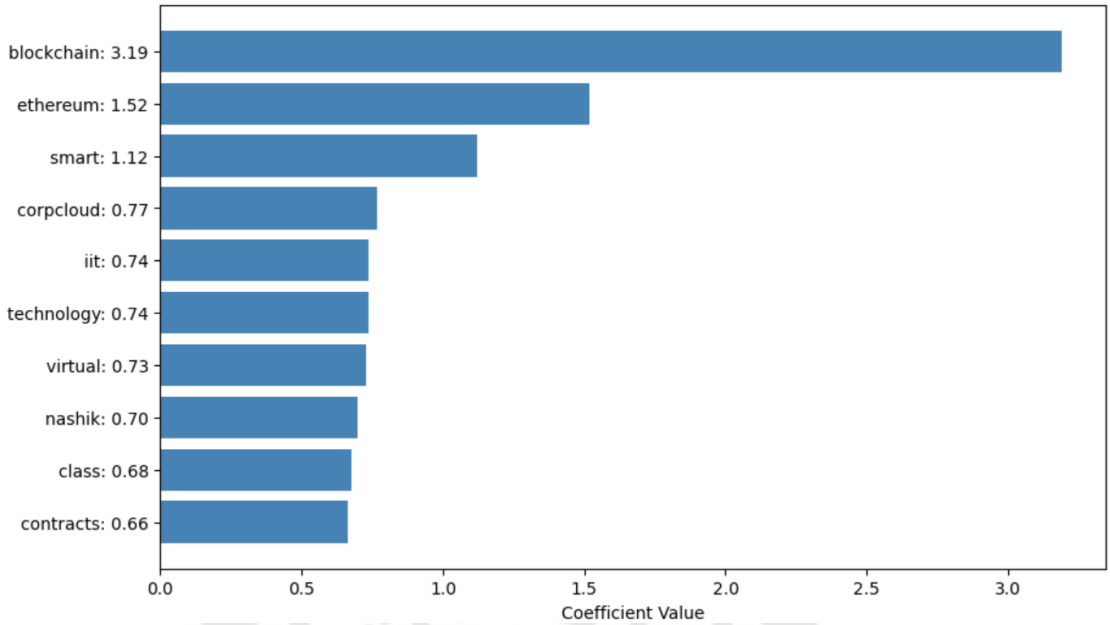
ภาพประกอบ 89 แสดงค่า *SHAP* ของ *Class 1 – Arts*



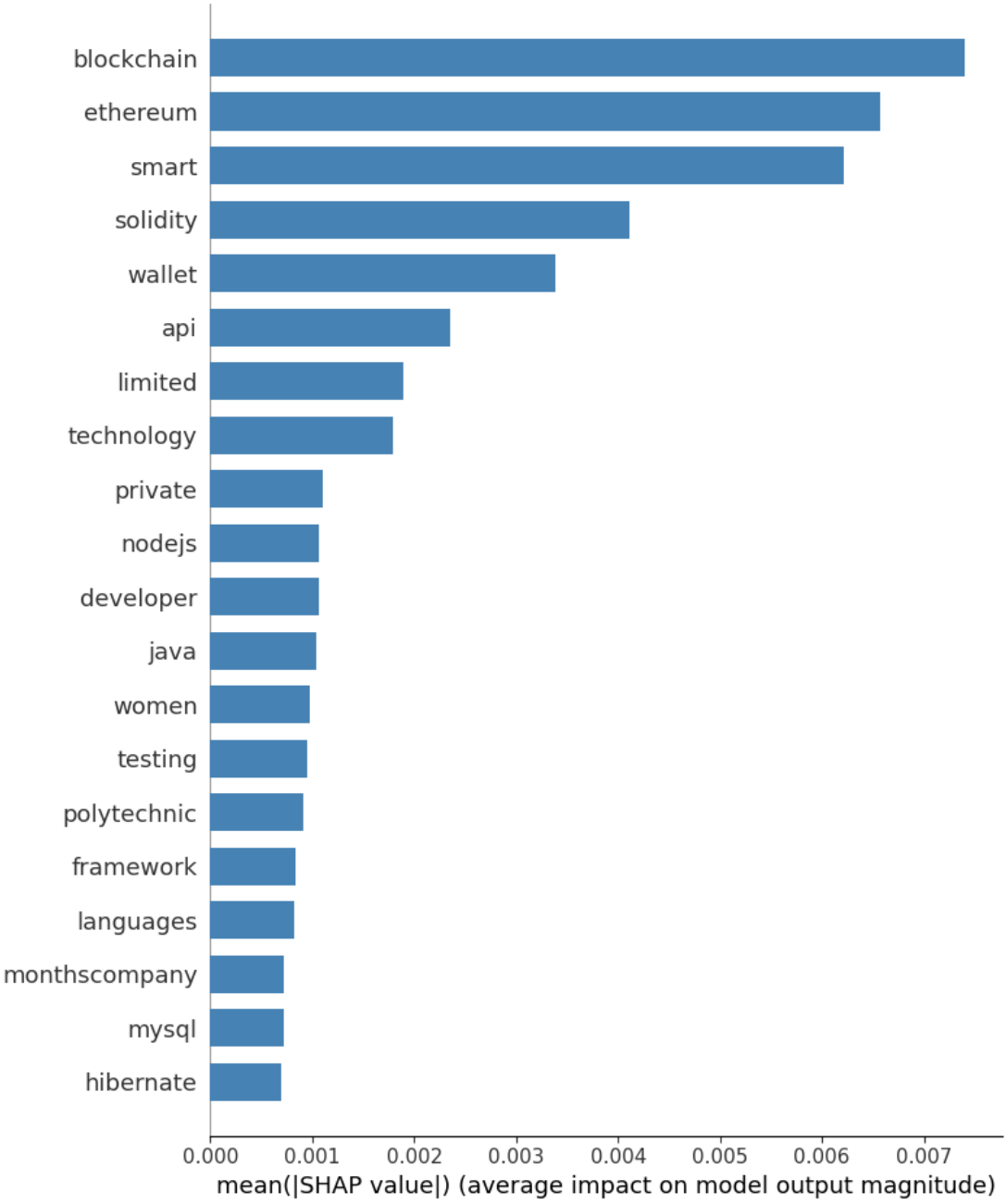
ภาพประกอบ 90 แสดงค่า *Feature Importance* ของ *Class 2 – Automation Testing*



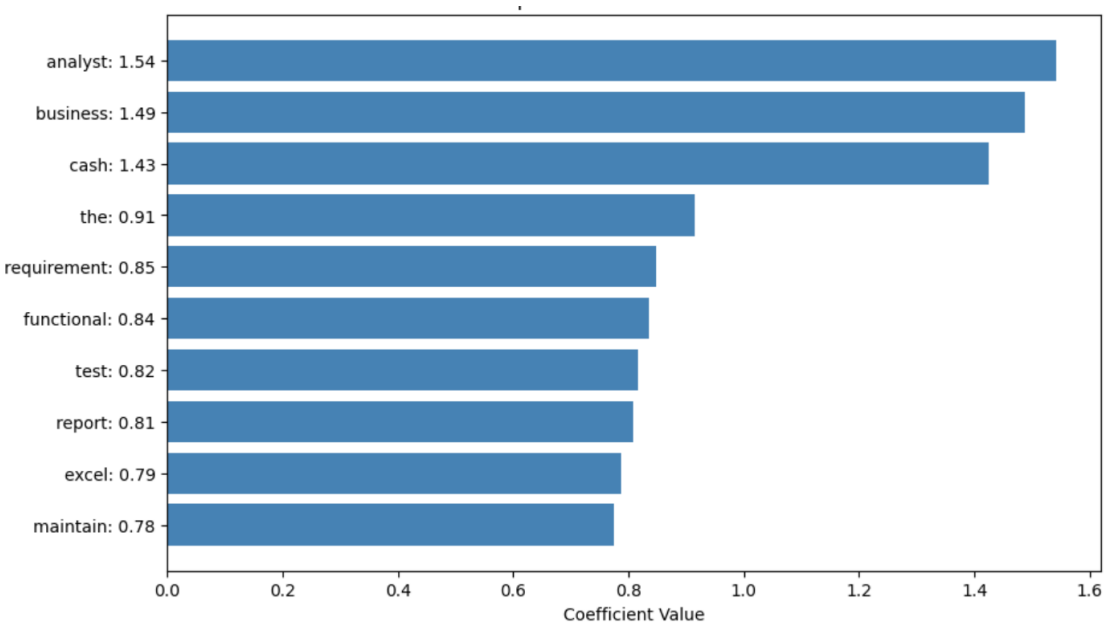
ภาพประกอบ 91 แสดงค่า *SHAP* ของ *Class 2 – Automation Testing*



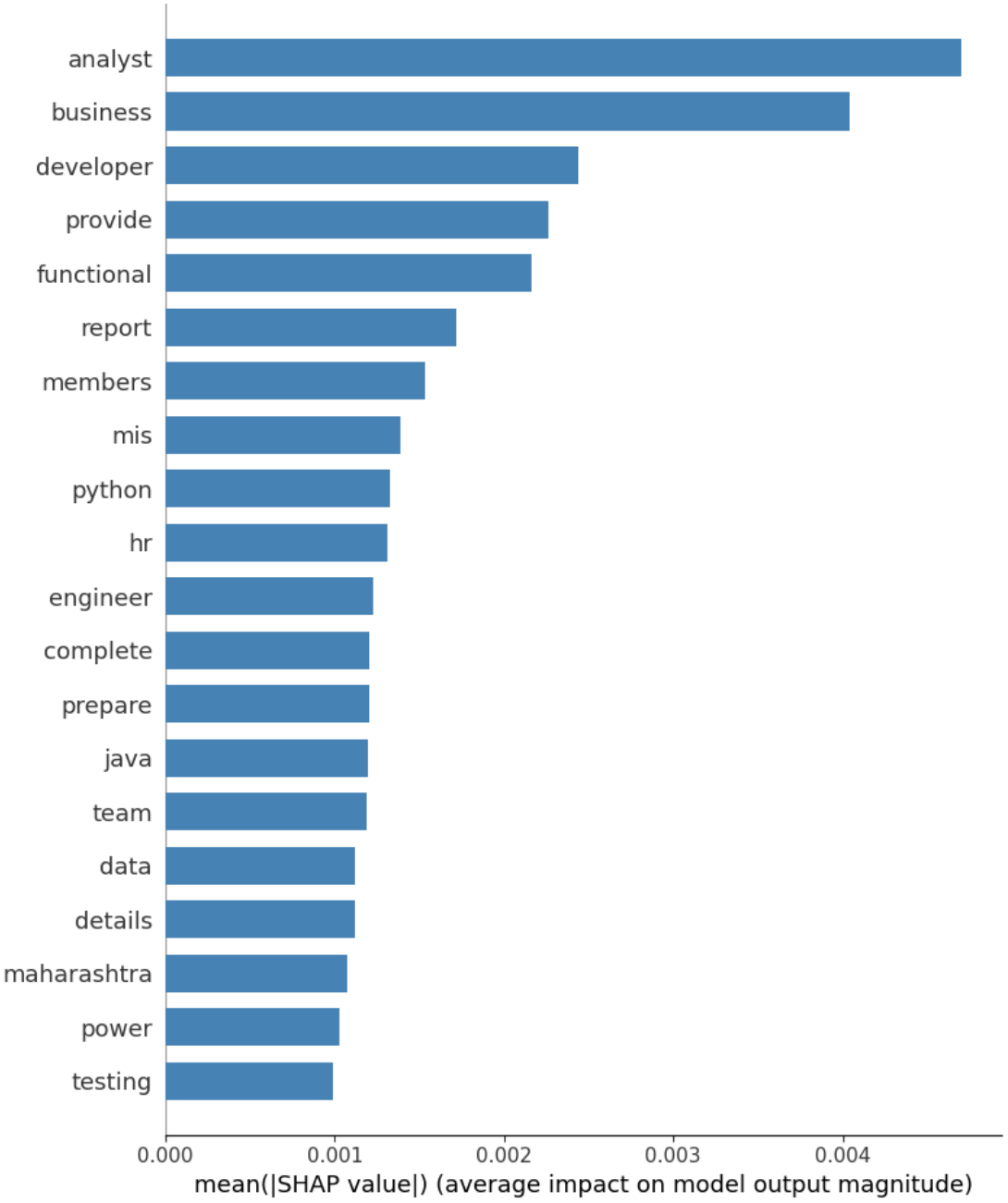
ภาพประกอบ 92 แสดงค่า *Feature Importance* ของ *Class 3 – Blockchain*



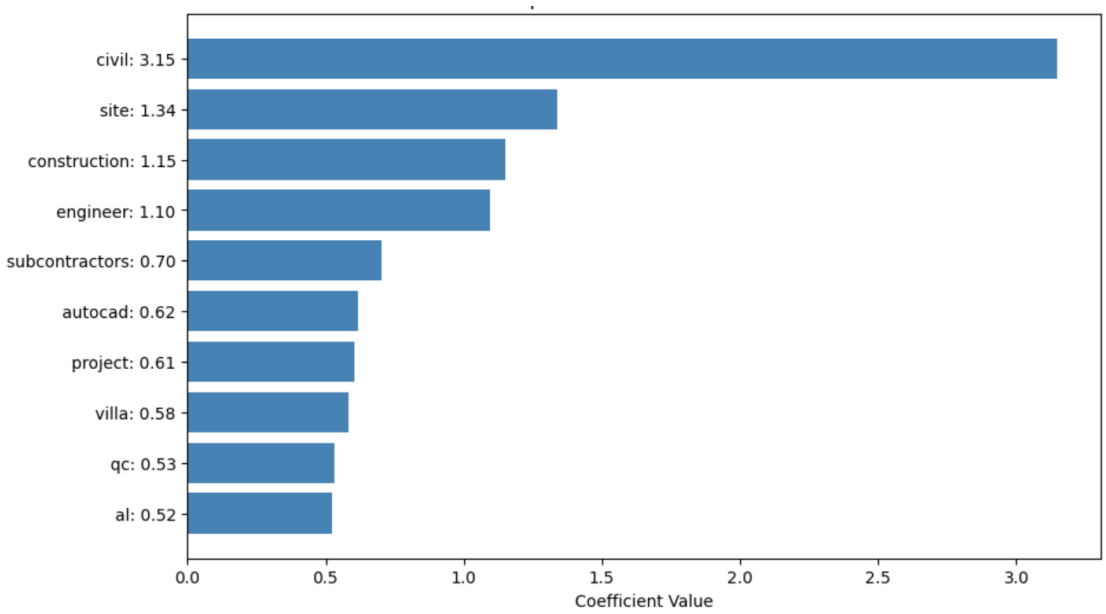
ภาพประกอบ 93 แสดงค่า *SHAP* ของ *Class 3 – Blockchain*



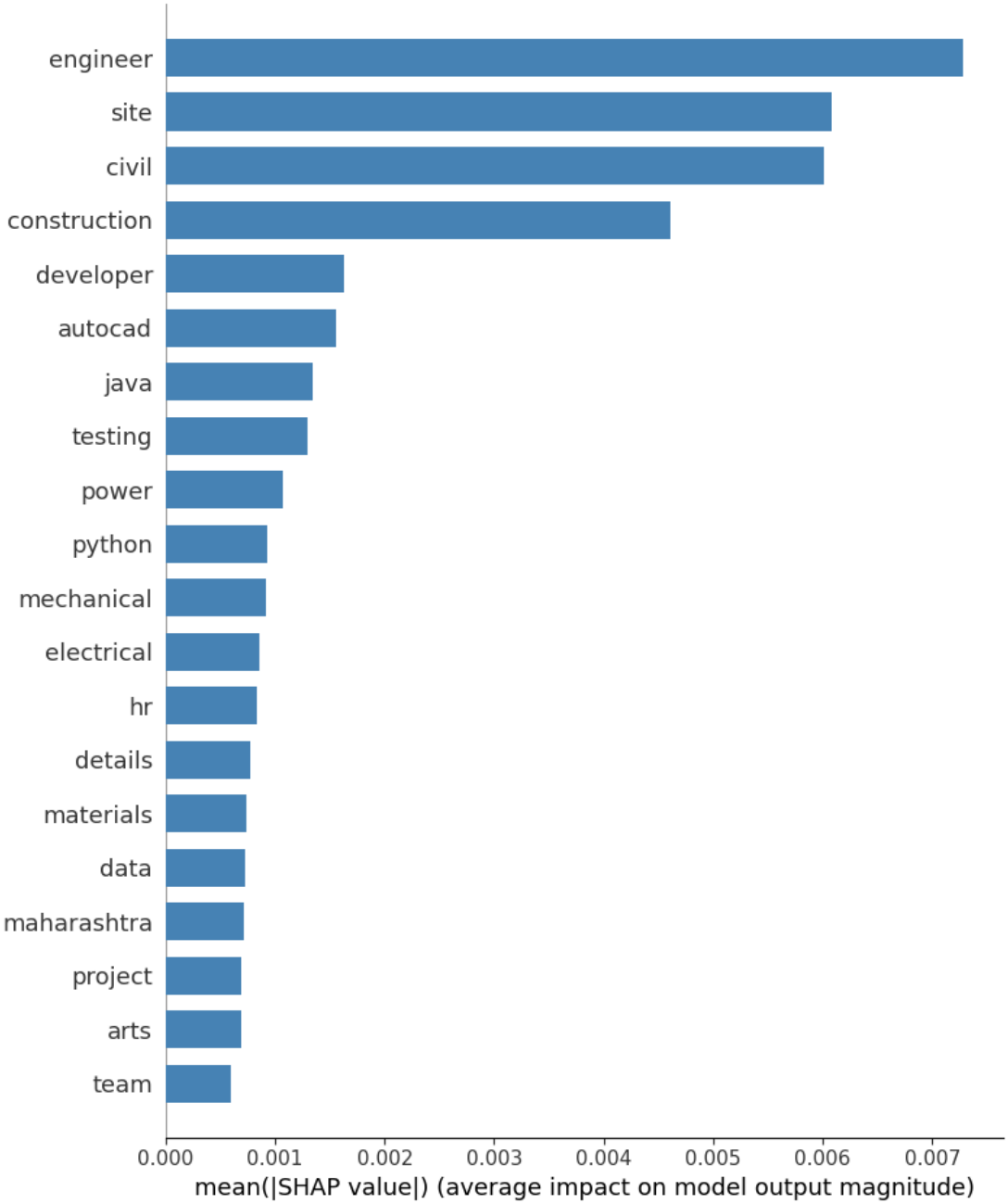
ภาพประกอบ 94 แสดงค่า *Feature Importance* ของ *Class 4 – Business Analyst*



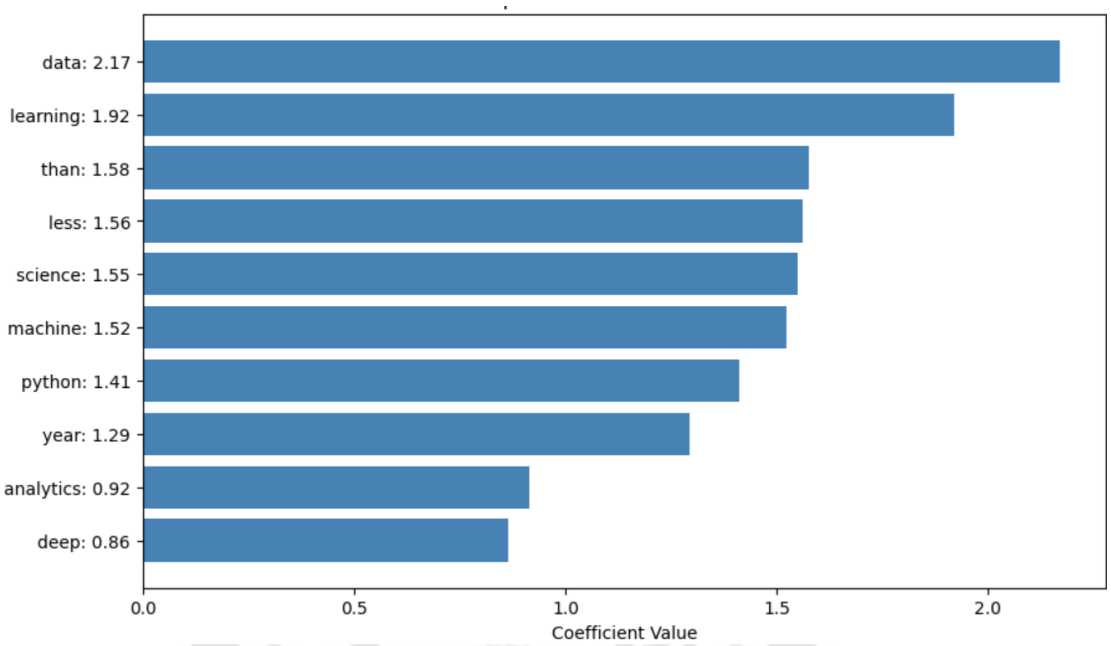
ภาพประกอบ 95 แสดงค่า *SHAP* ของ *Class 4 – Business Analyst*



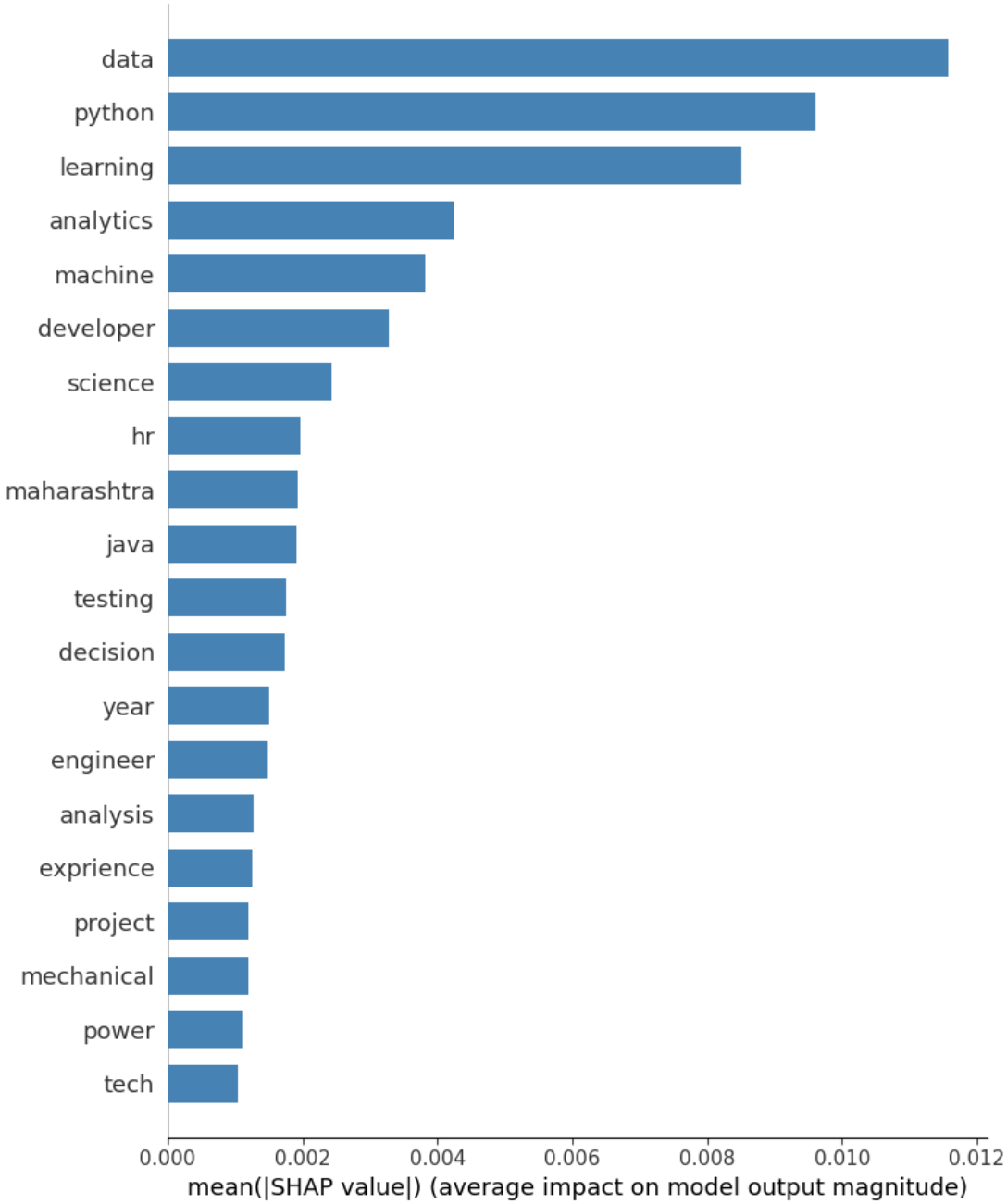
ภาพประกอบ 96 แสดงค่า *Feature Importance* ของ *Class 5 – Civil Engineer*



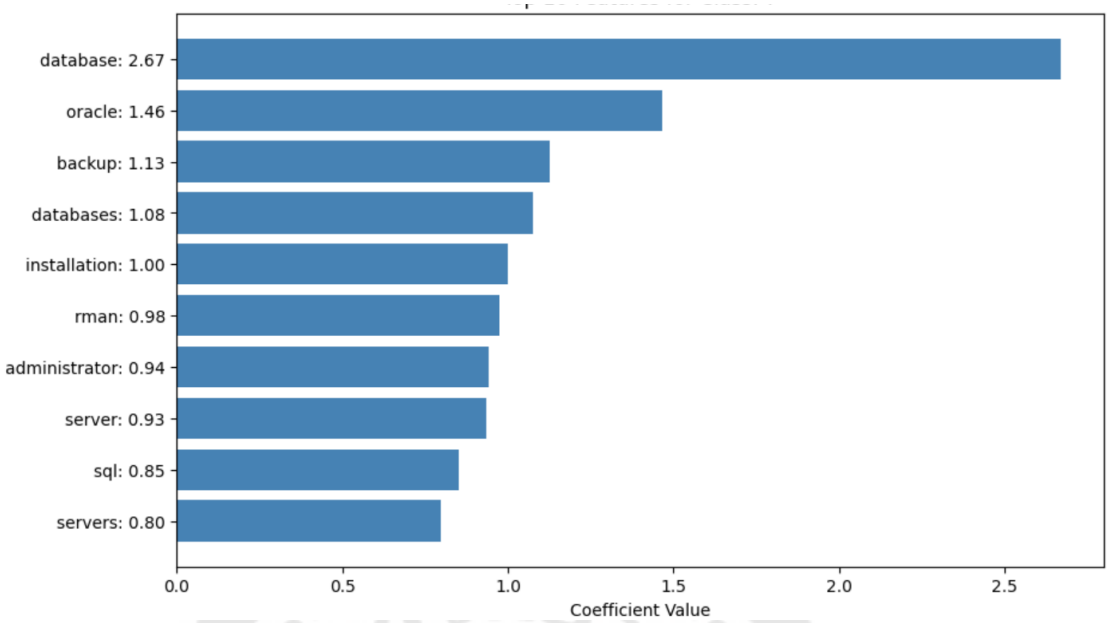
ภาพประกอบ 97 แสดงค่า *SHAP* ของ *Class 5 – Civil Engineer*



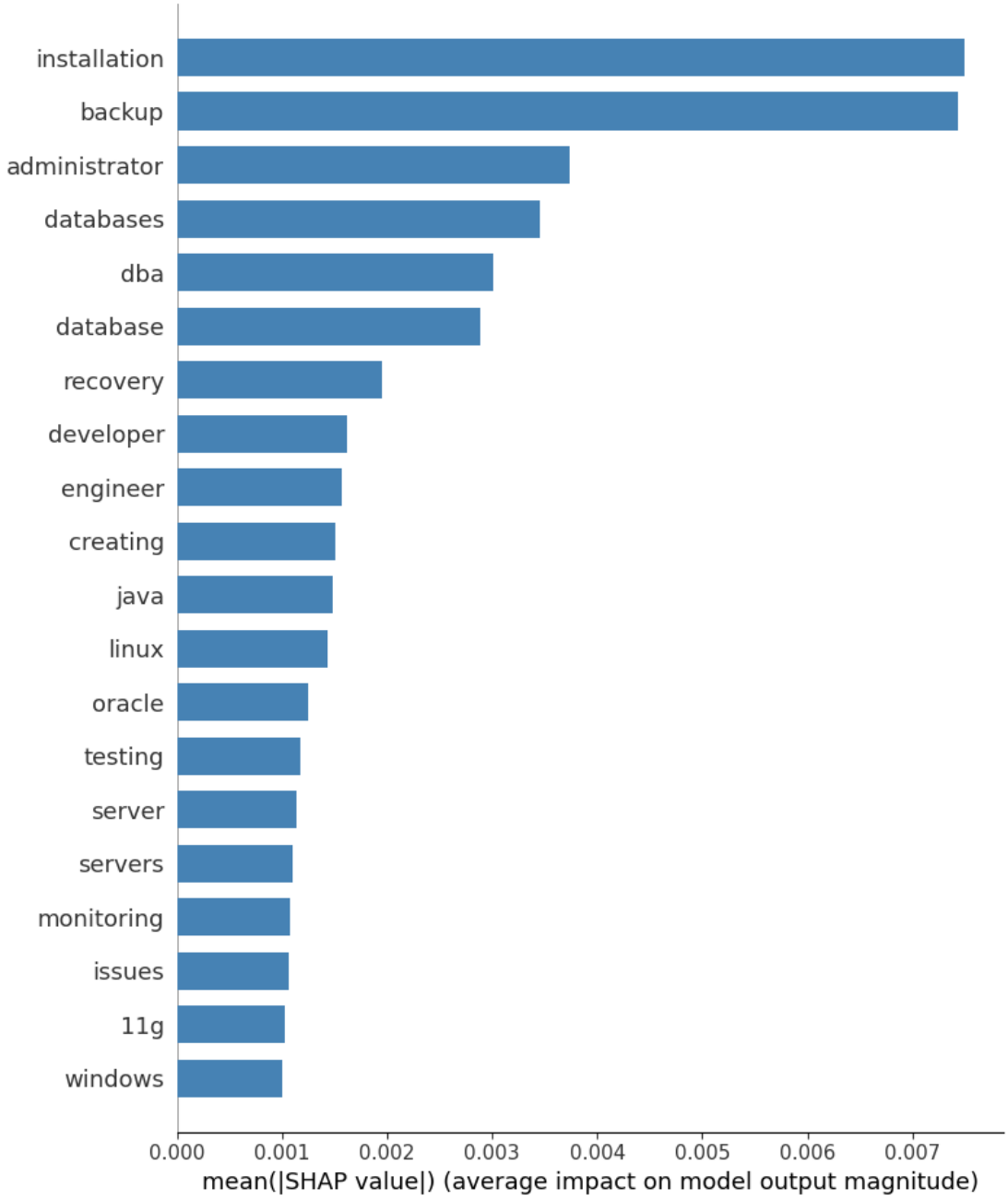
ภาพประกอบ 98 แสดงค่า *Feature Importance* ของ *Class 6 – Data Science*



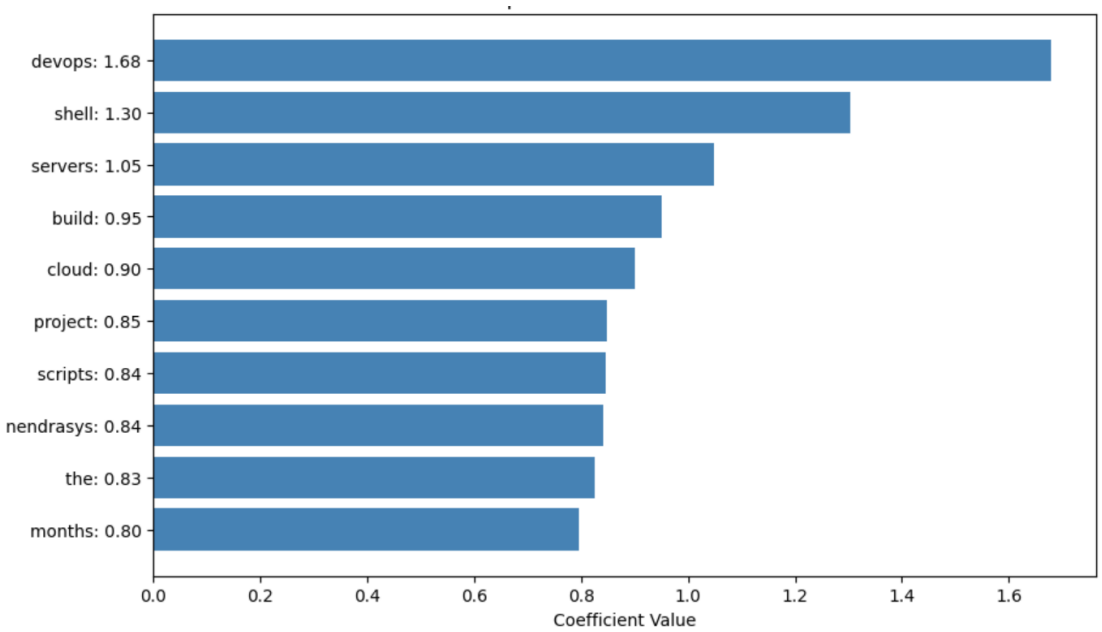
ภาพประกอบ 99 แสดงค่า *SHAP* ของ *Class 6 – Data Science*



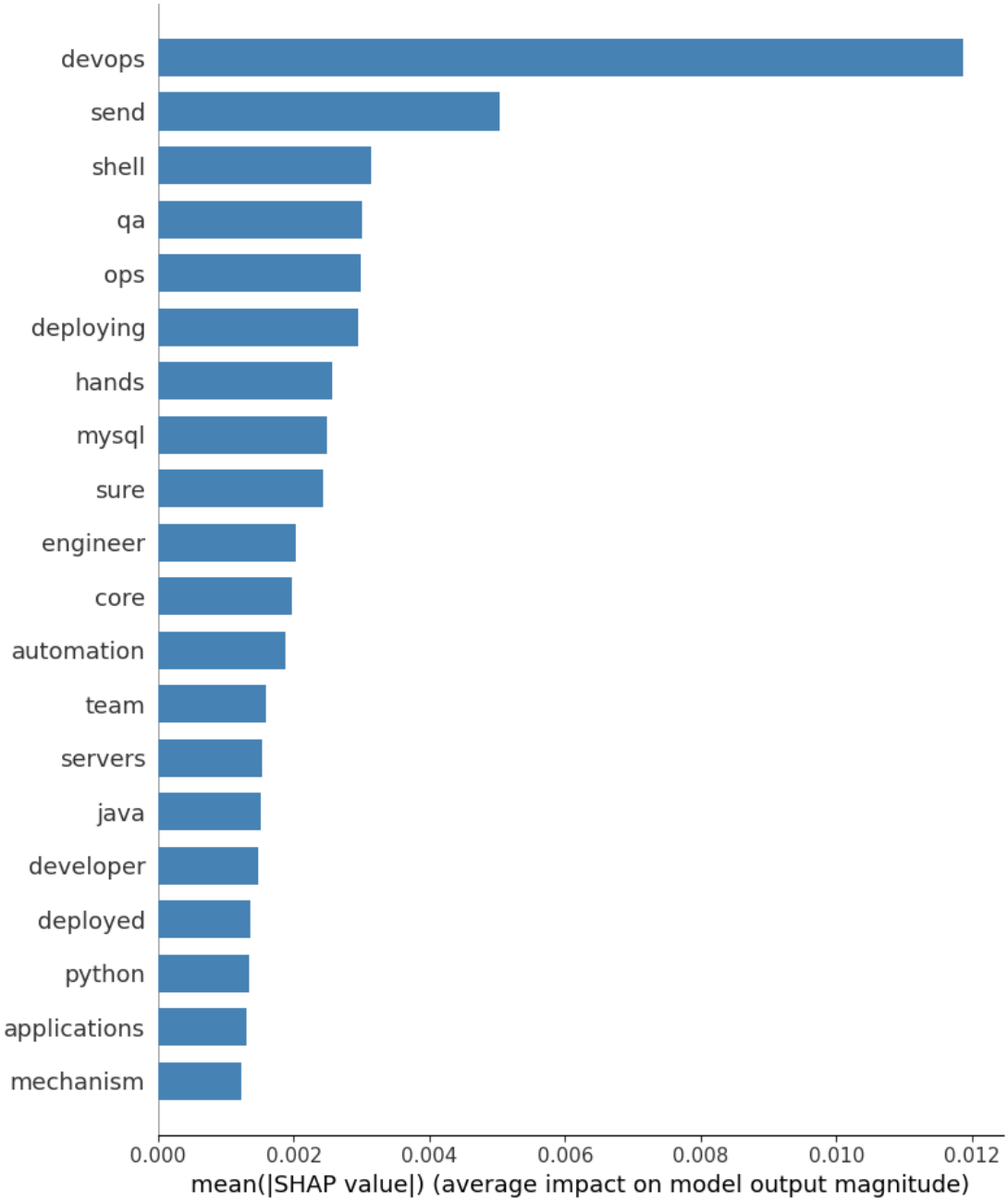
ภาพประกอบ 100 แสดงค่า *Feature Importance* ของ *Class 7 – Database*



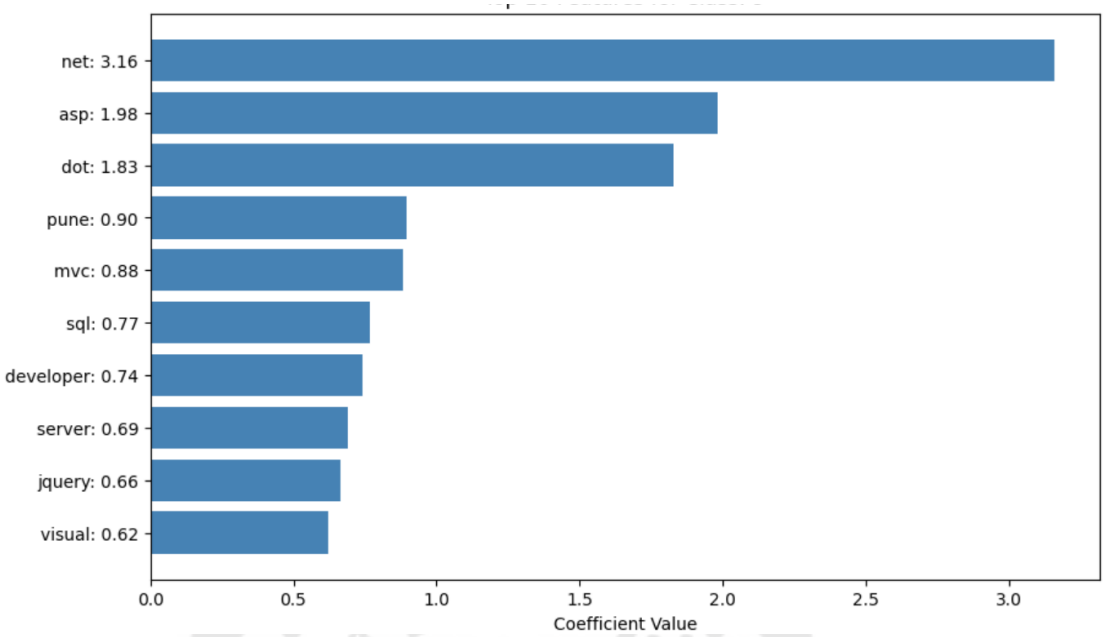
ภาพประกอบ 101 แสดงค่า *SHAP* ของ *Class 7 – Database*



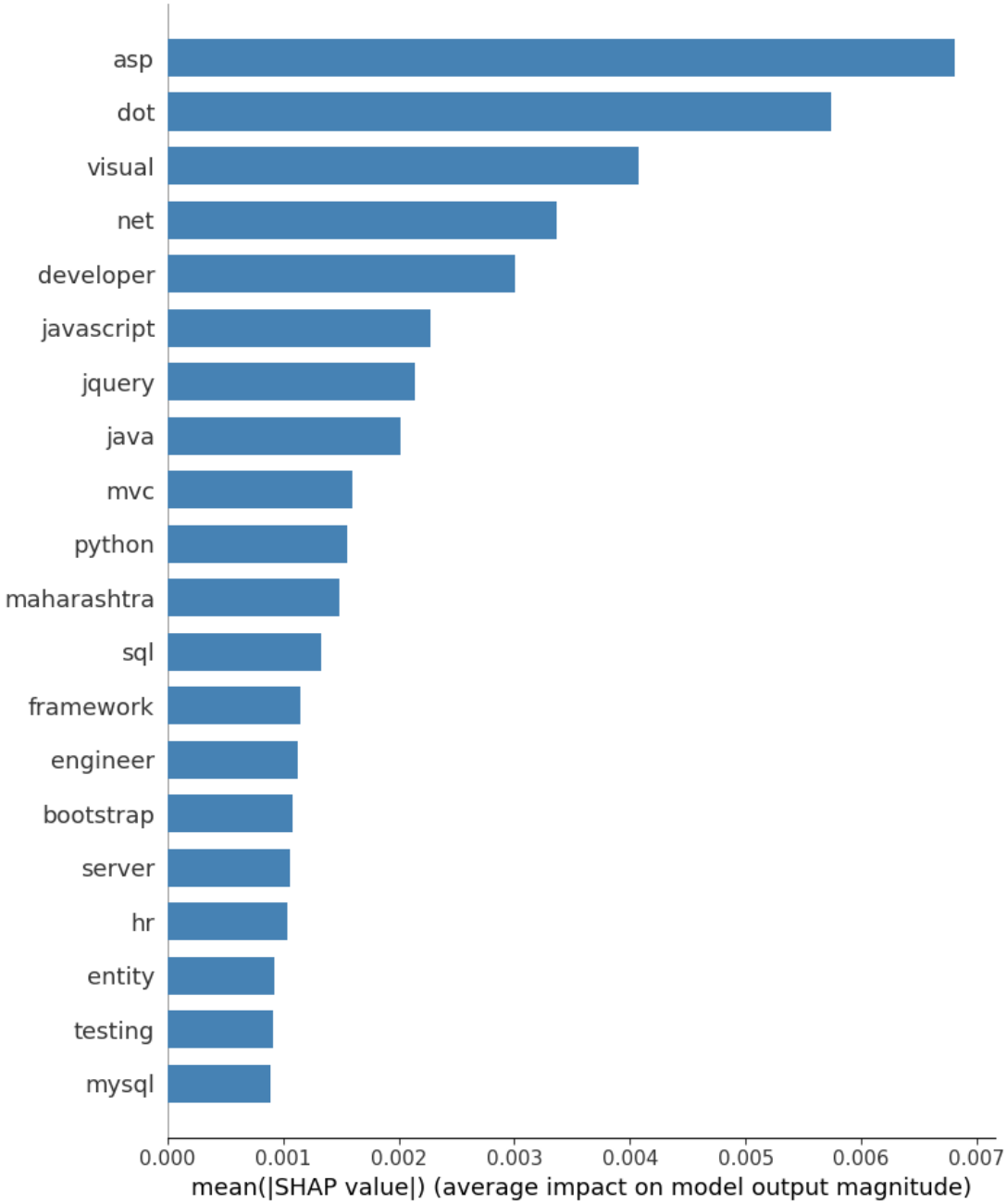
ภาพประกอบ 102 แสดงค่า *Feature Importance* ของ *Class 8 – DevOps Engineer*



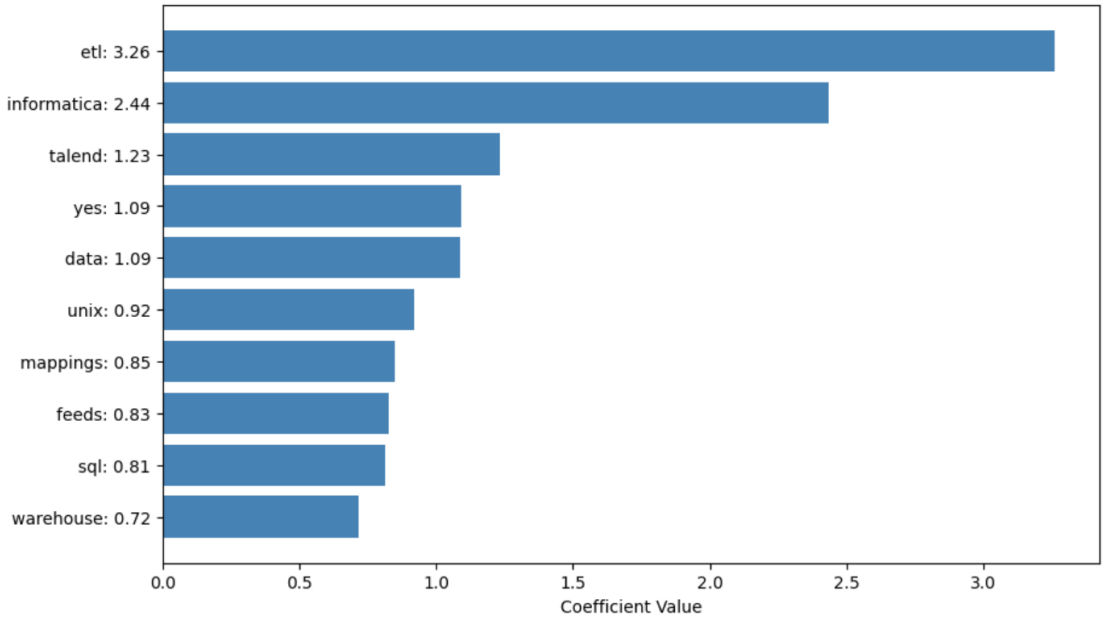
ภาพประกอบ 103 แสดงค่า *SHAP* ของ *Class 8 – DevOps Engineer*



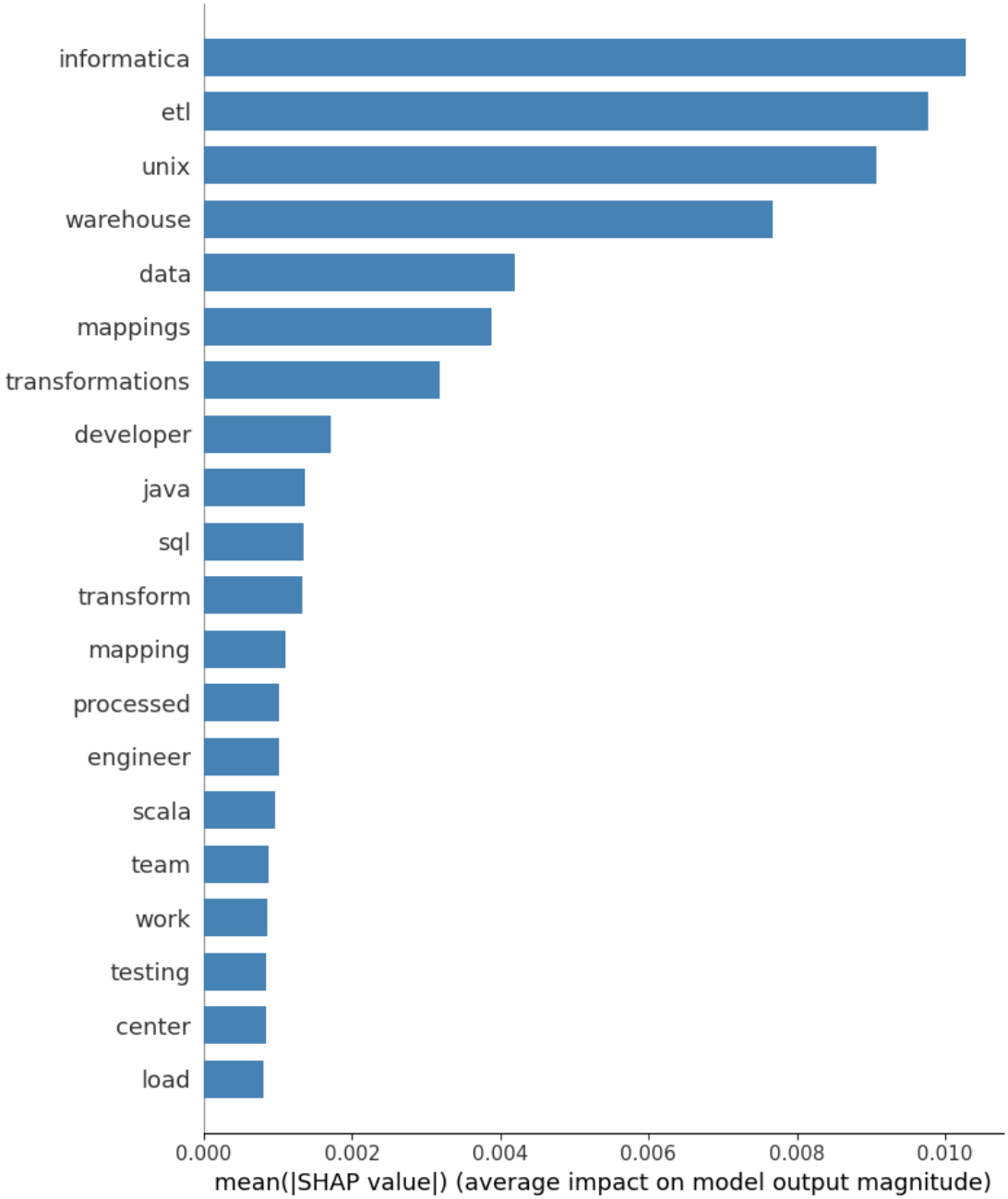
ภาพประกอบ 104 แสดงค่า *Feature Importance* ของ *Class 9 – DotNet Developer*



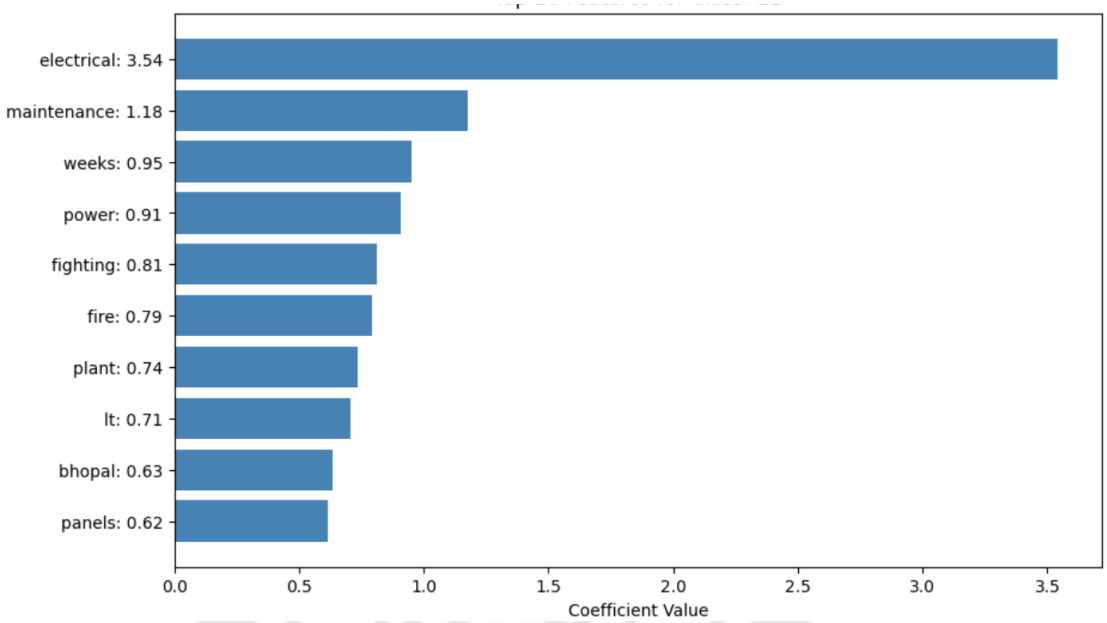
ภาพประกอบ 105 แสดงค่า *SHAP* ของ *Class 9 – DotNet Developer*



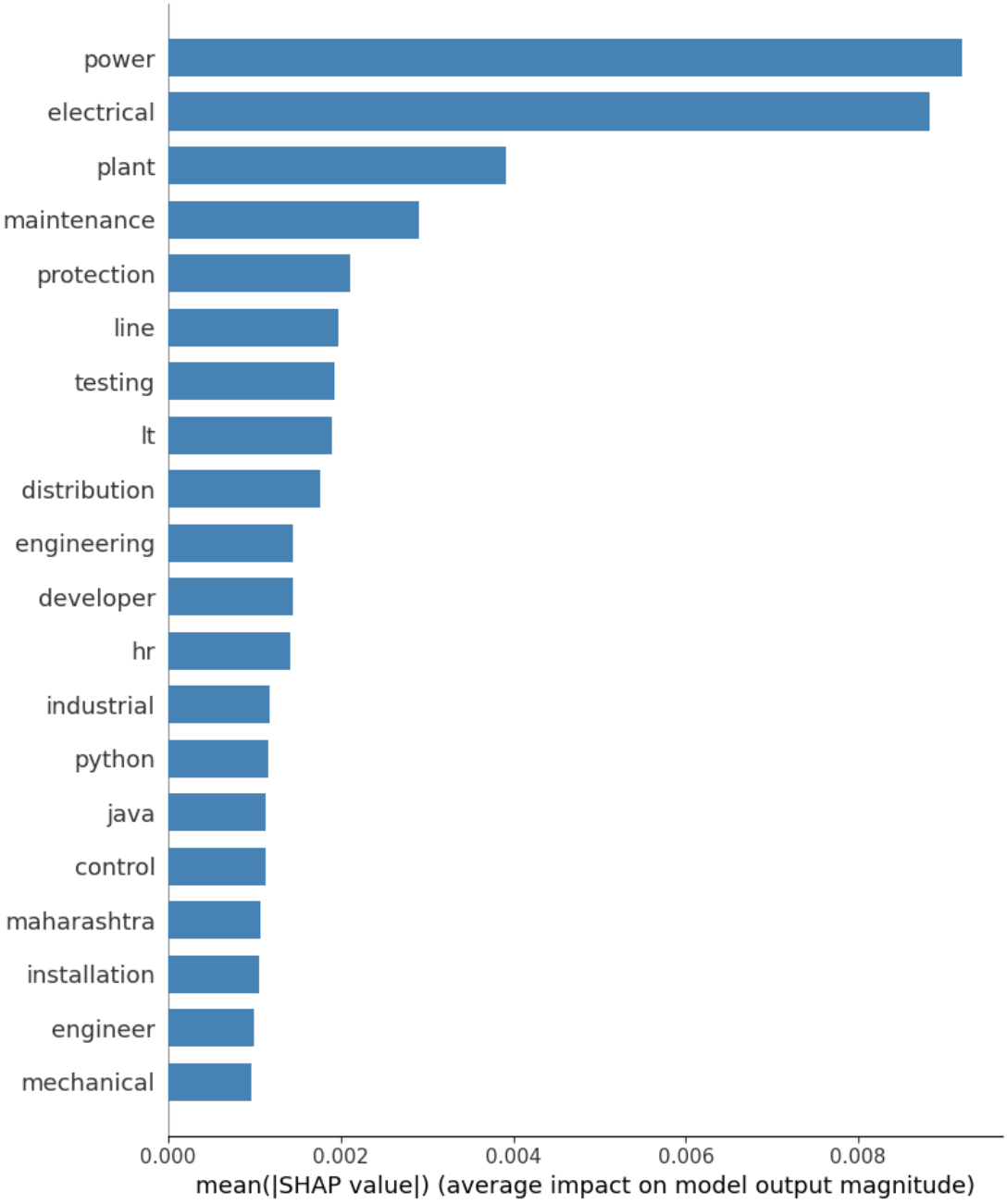
ภาพประกอบ 106 แสดงค่า *Feature Importance* ของ *Class 10 – ETL Developer*



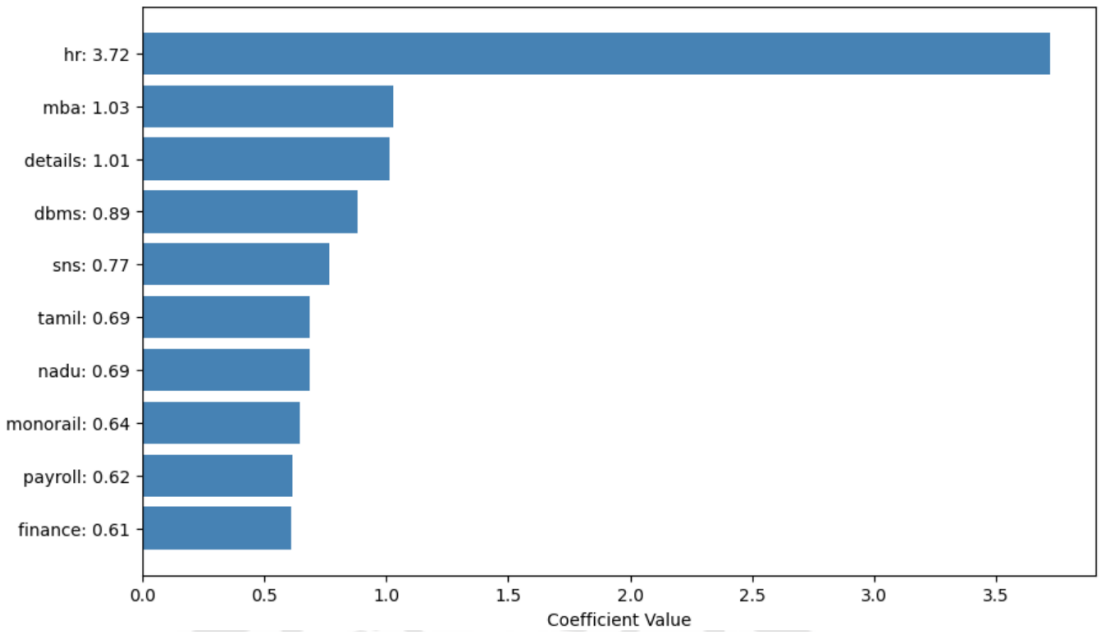
ภาพประกอบ 107 แสดงค่า *SHAP* ของ *Class 10 – ETL Developer*



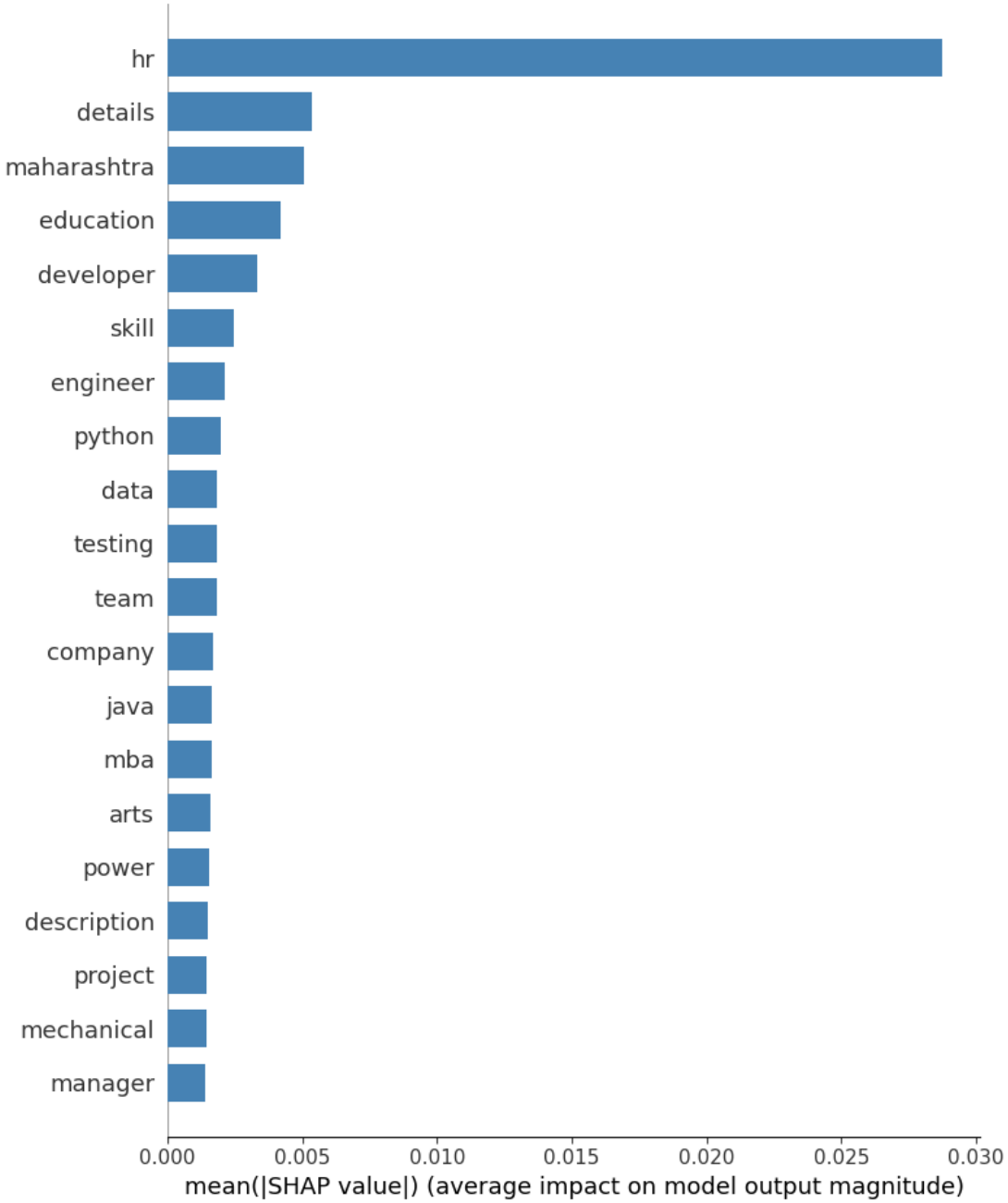
ภาพประกอบ 108 แสดงค่า *Feature Importance* ของ *Class 11 – Electrical Engineering*



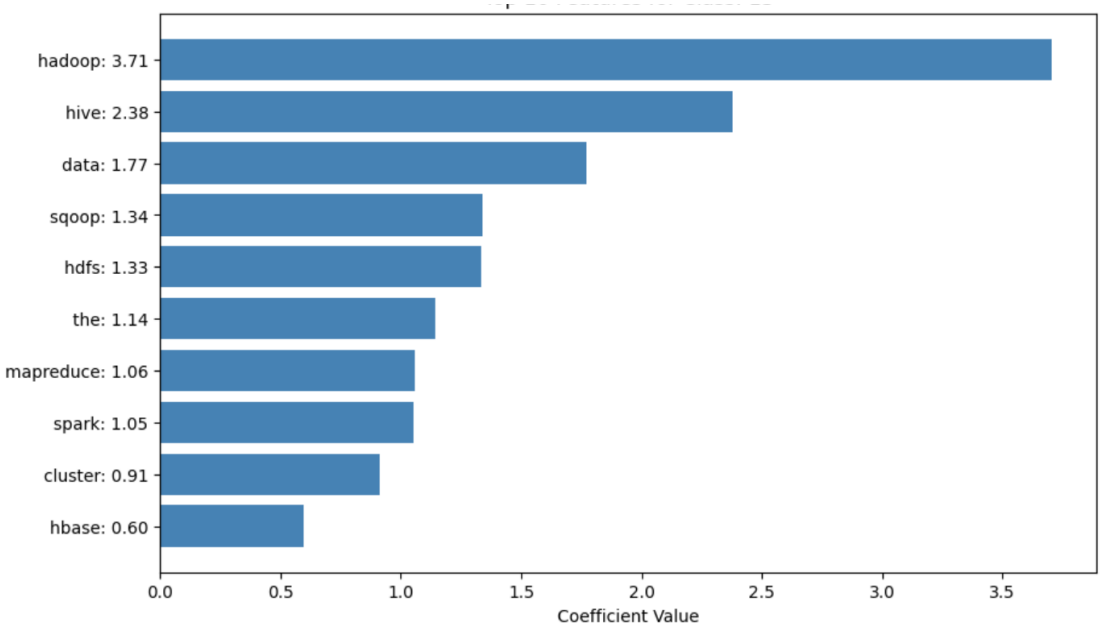
ภาพประกอบ 109 แสดงค่า *SHAP* ของ *Class 11 – Electrical Engineering*



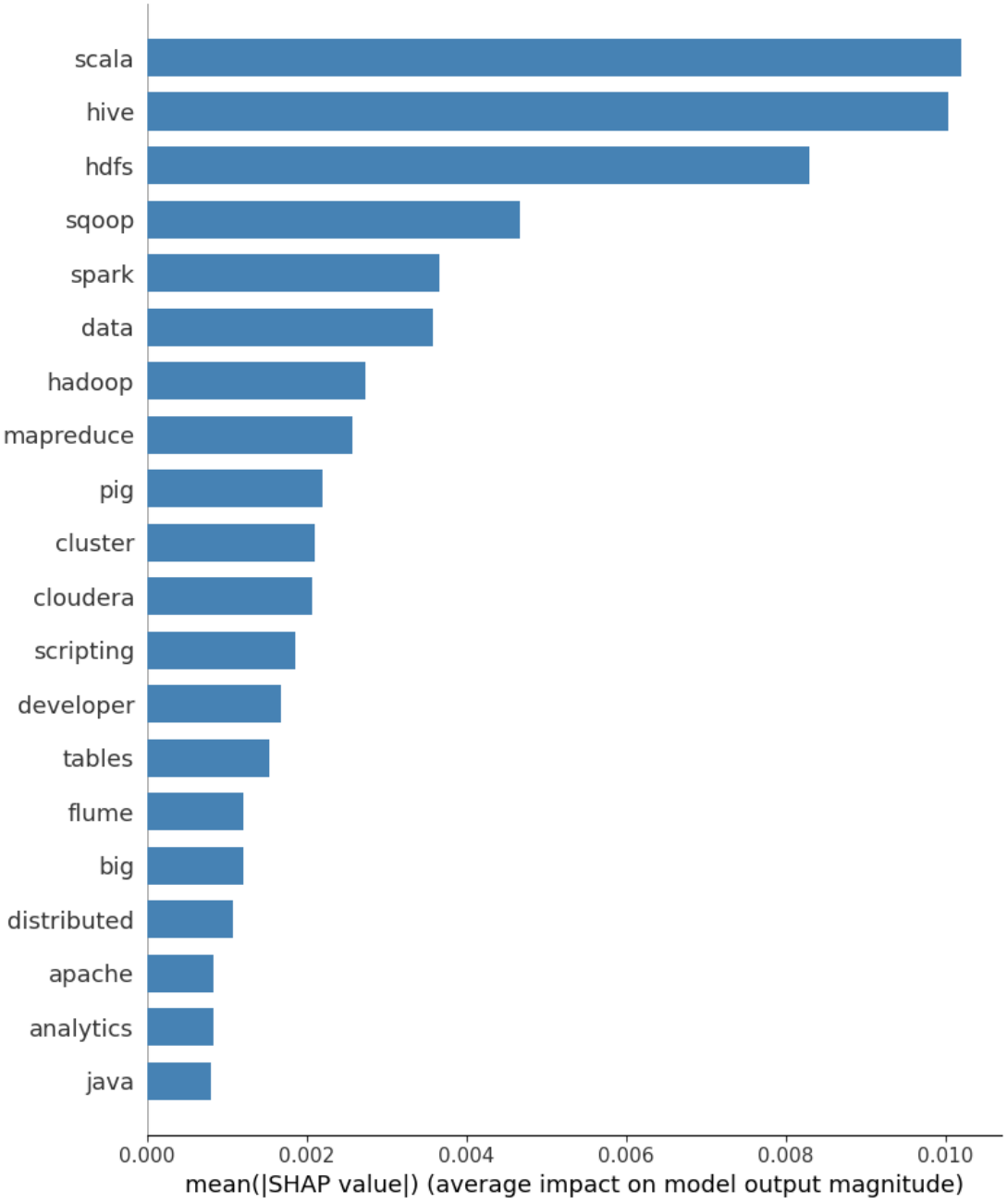
ภาพประกอบ 110 แสดงค่า *Feature Importance* ของ *Class 12 – HR*



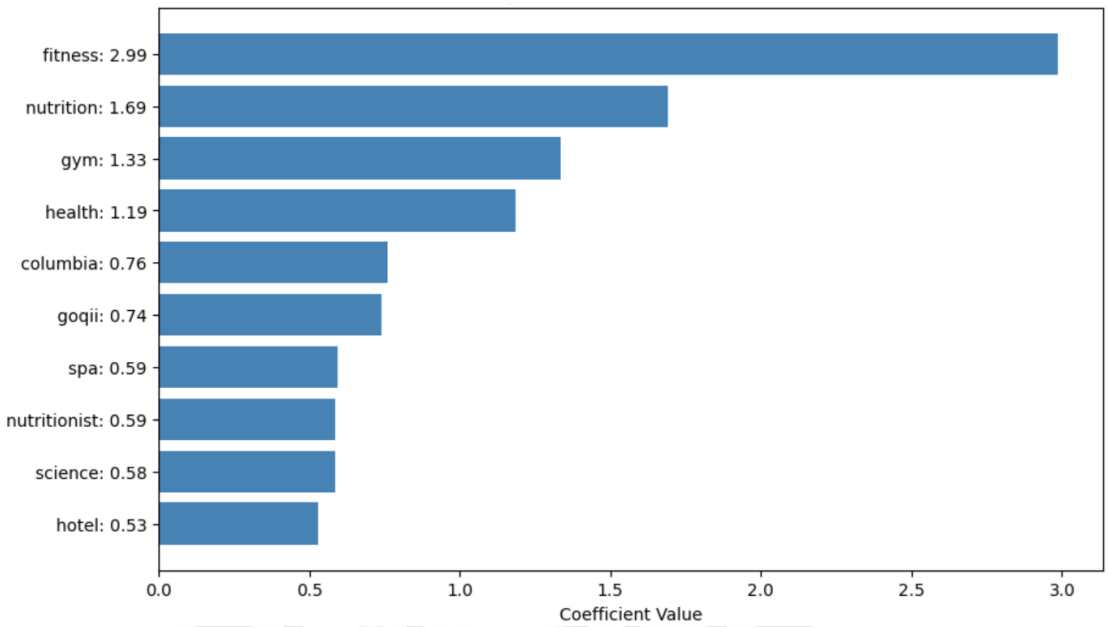
ภาพประกอบ 111 แสดงค่า *SHAP* ของ *Class 12 – HR*



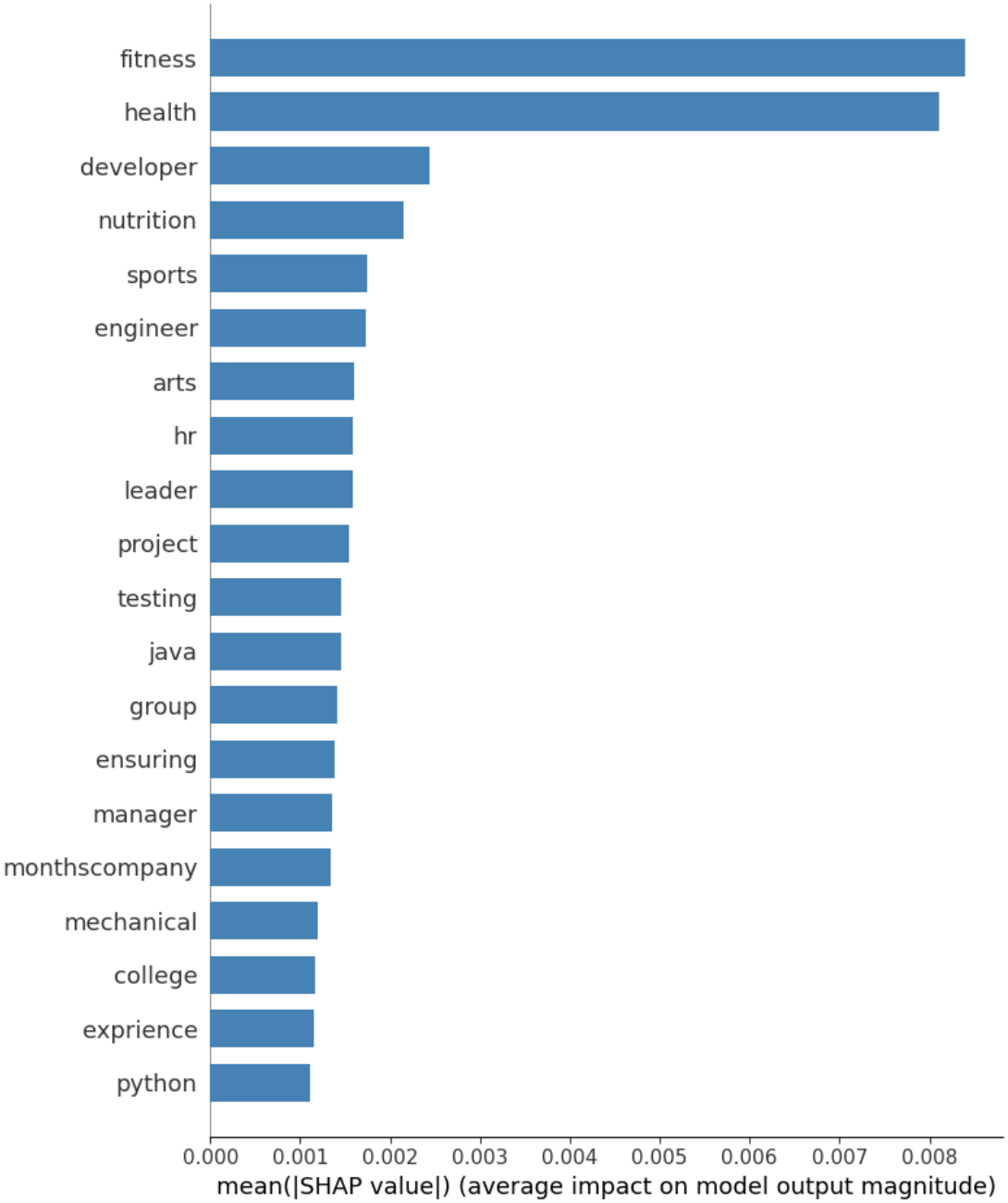
ภาพประกอบ 112 แสดงค่า *Feature Importance* ของ *Class 13 – Hadoop*



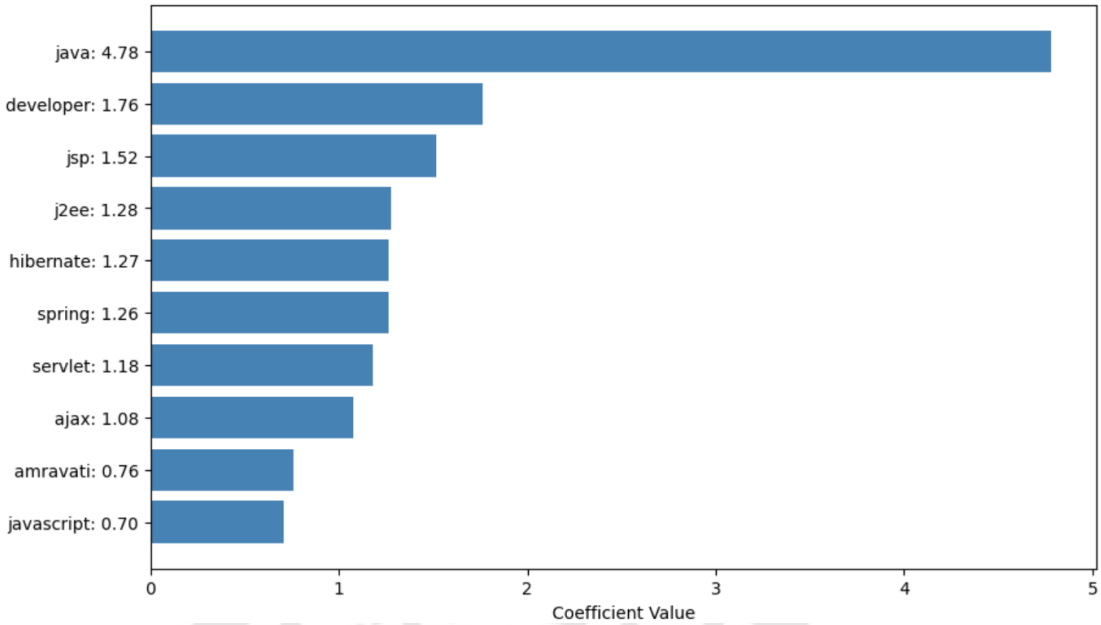
ภาพประกอบ 113 แสดงค่า *SHAP* ของ *Class 13 – Hadoop*



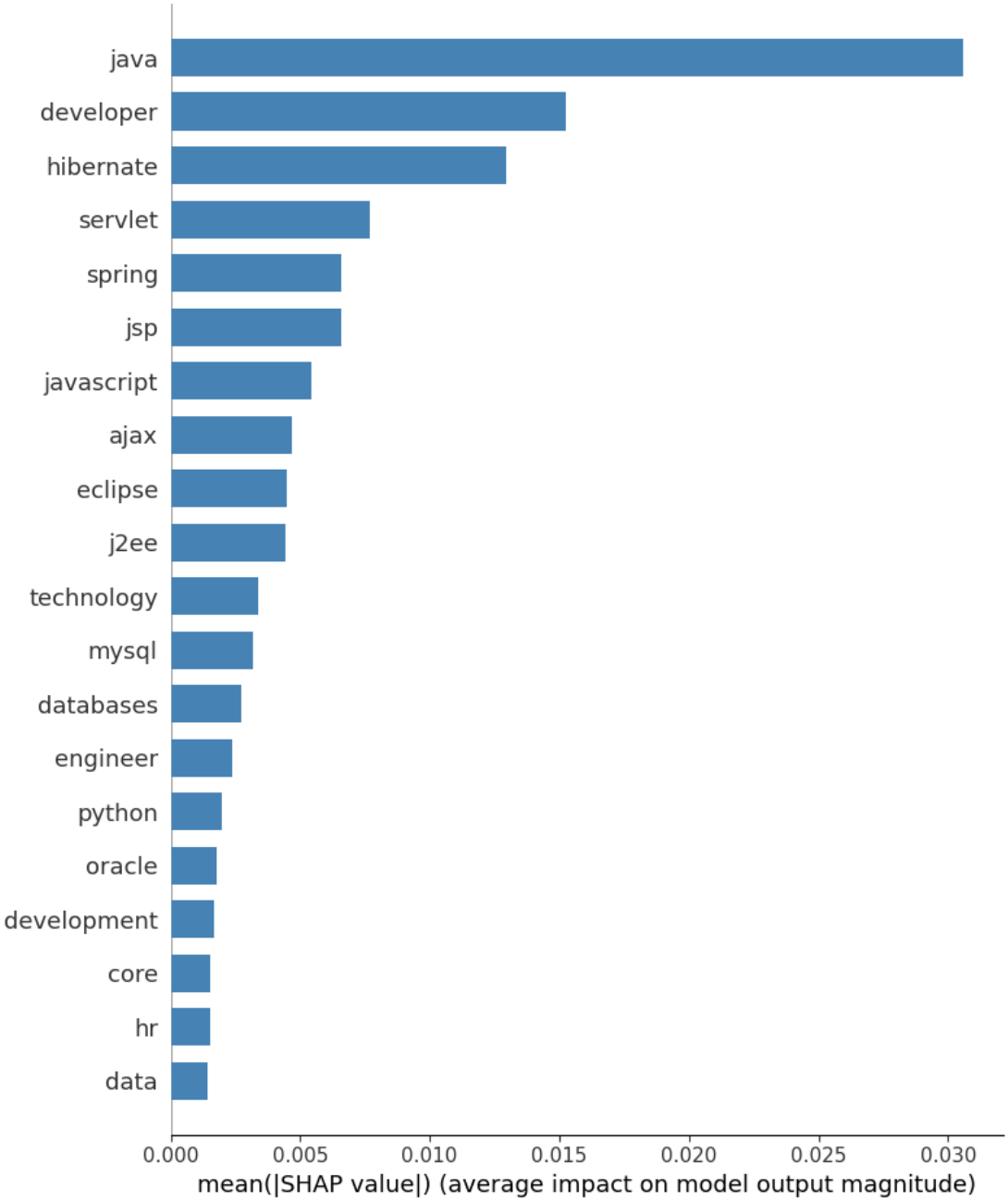
ภาพประกอบ 114 แสดงค่า *Feature Importance* ของ *Class 14 – Health and fitness*



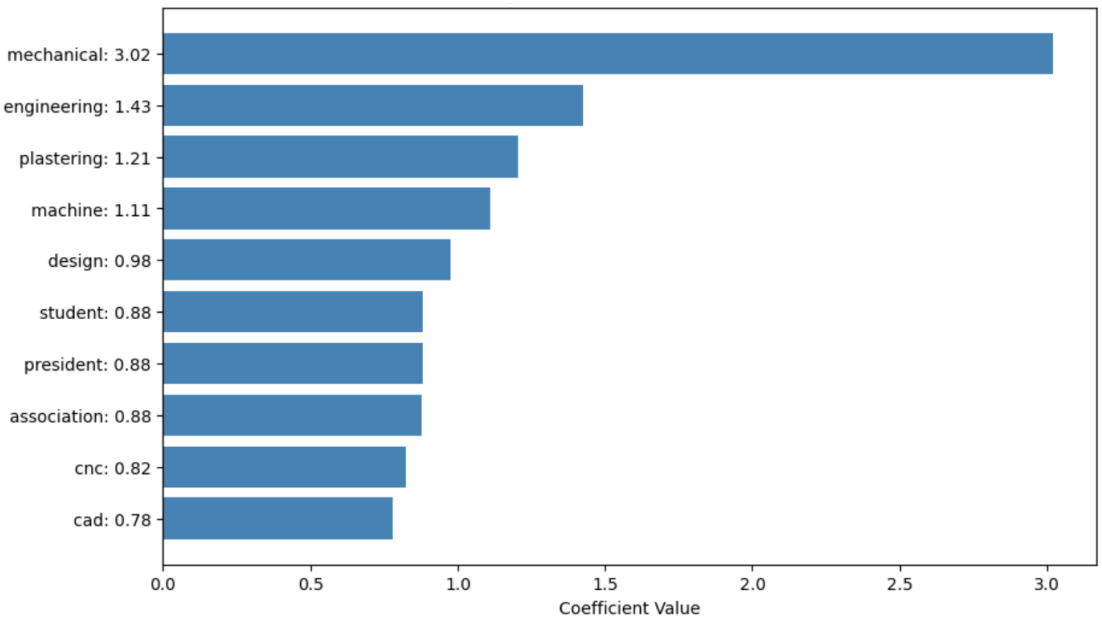
ภาพประกอบ 115 แสดงค่า *SHAP* ของ *Class 14 – Health and fitness*



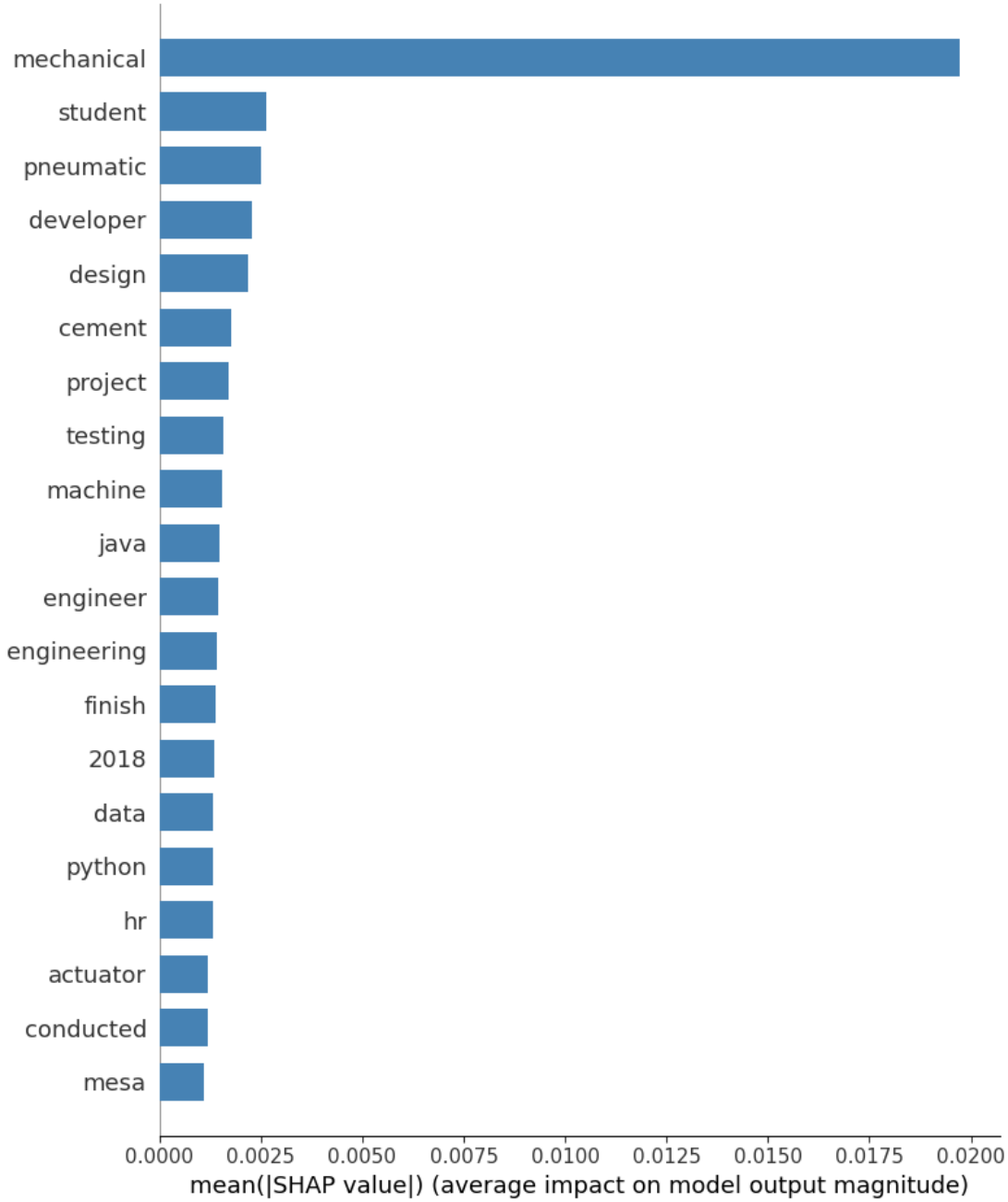
ภาพประกอบ 116 แสดงค่า *Feature Importance* ของ *Class 15 – Java Developer*



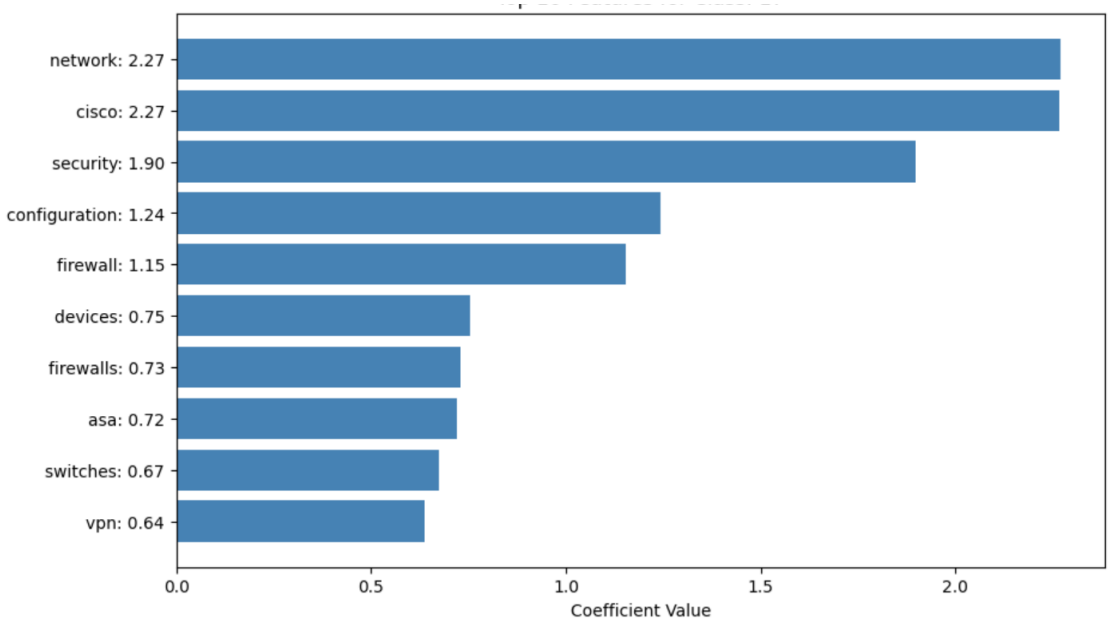
ภาพประกอบ 117แสดงค่า *SHAP* ของ *Class 15 – Java Developer*



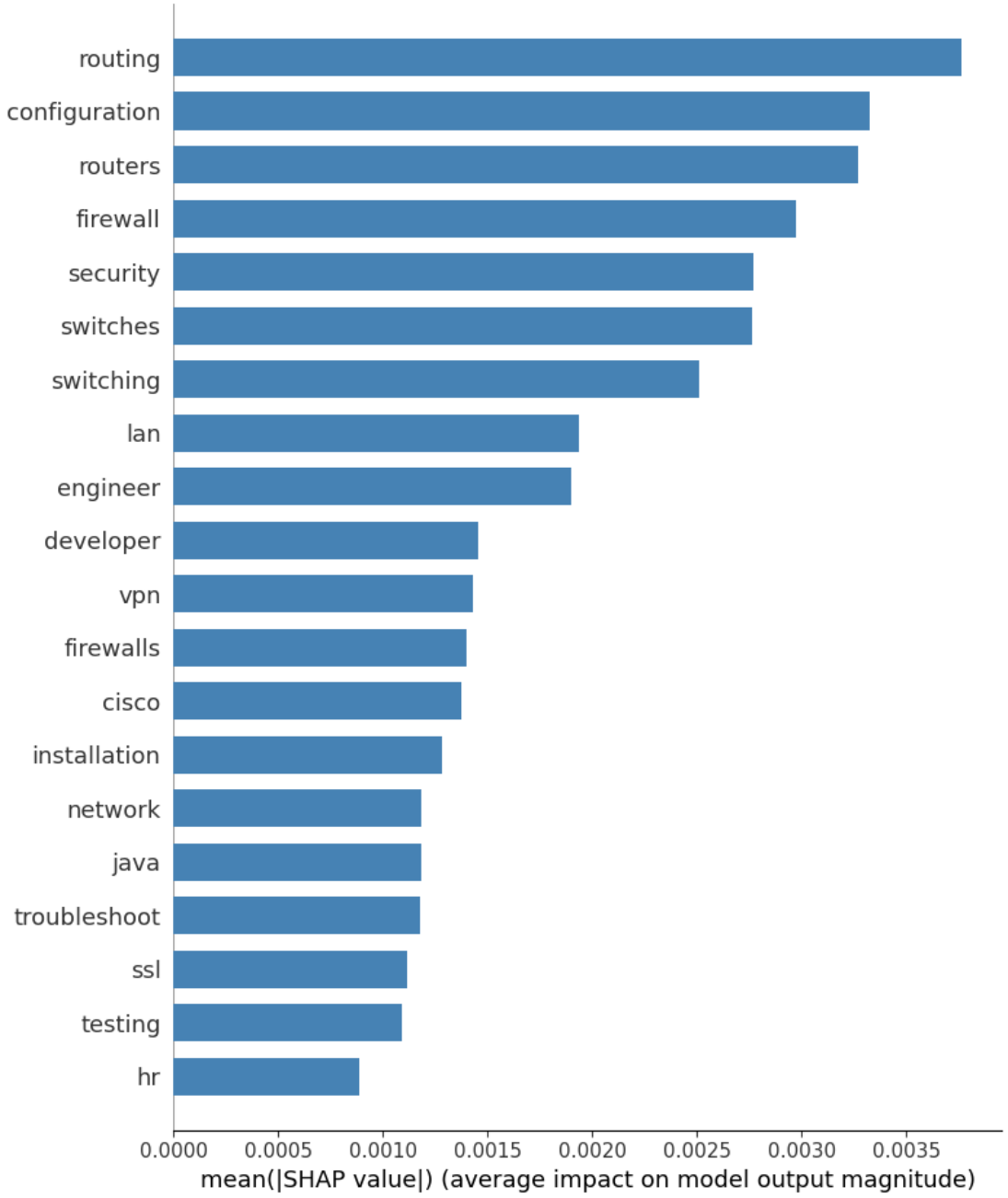
ภาพประกอบ 118 แสดงค่า *Feature Importance* ของ *Class 16 – Mechanical Engineer*



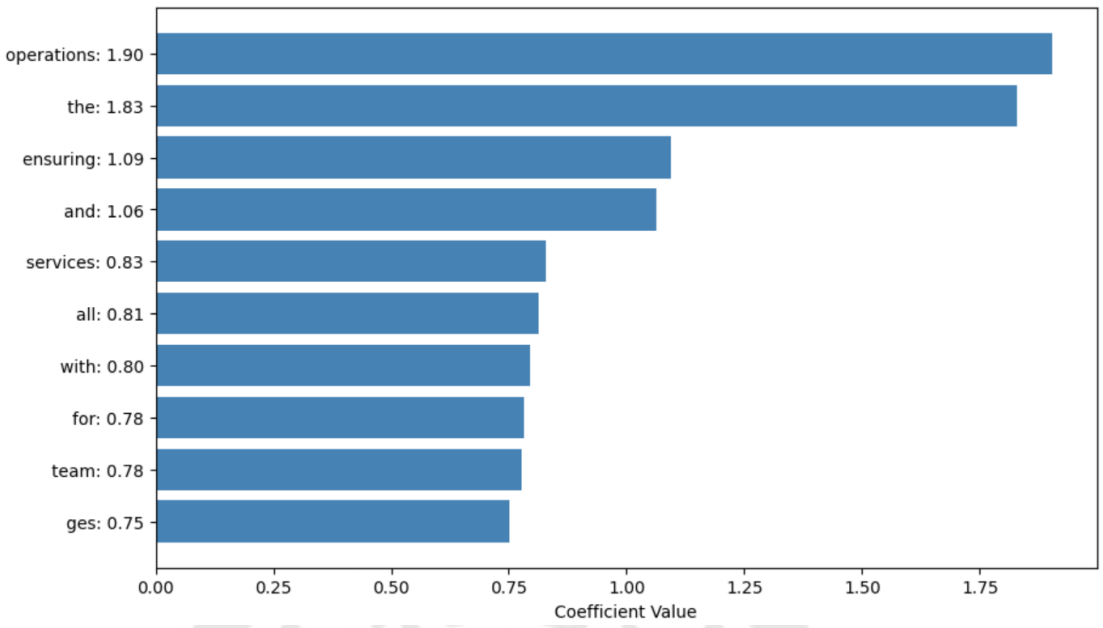
ภาพประกอบ 119 แสดงค่า *SHAP* ของ *Class 16 – Mechanical Engineer*



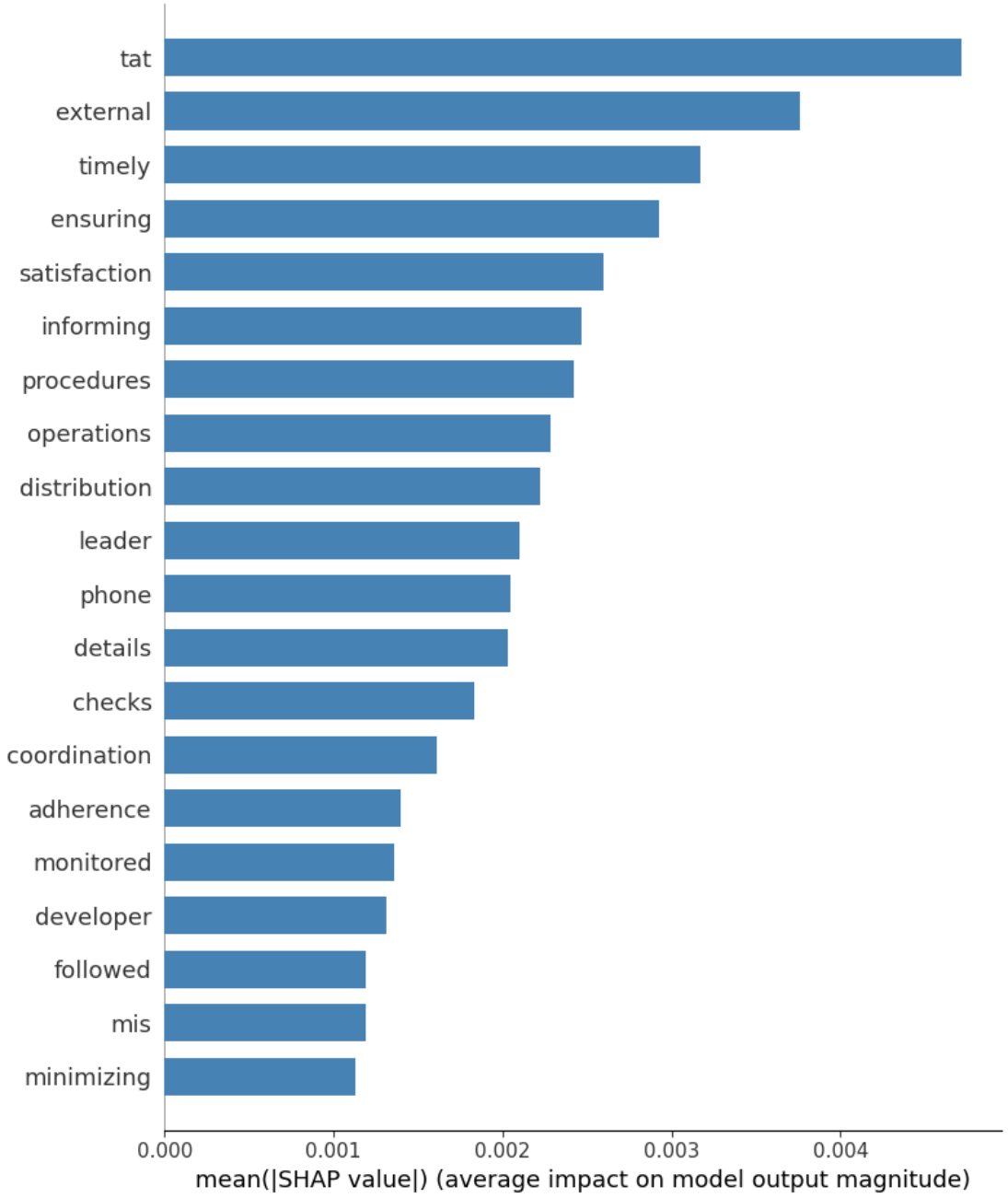
ภาพประกอบ 120 แสดงค่า *Feature Importance* ของ *Class 17 – Network Security Engineer*



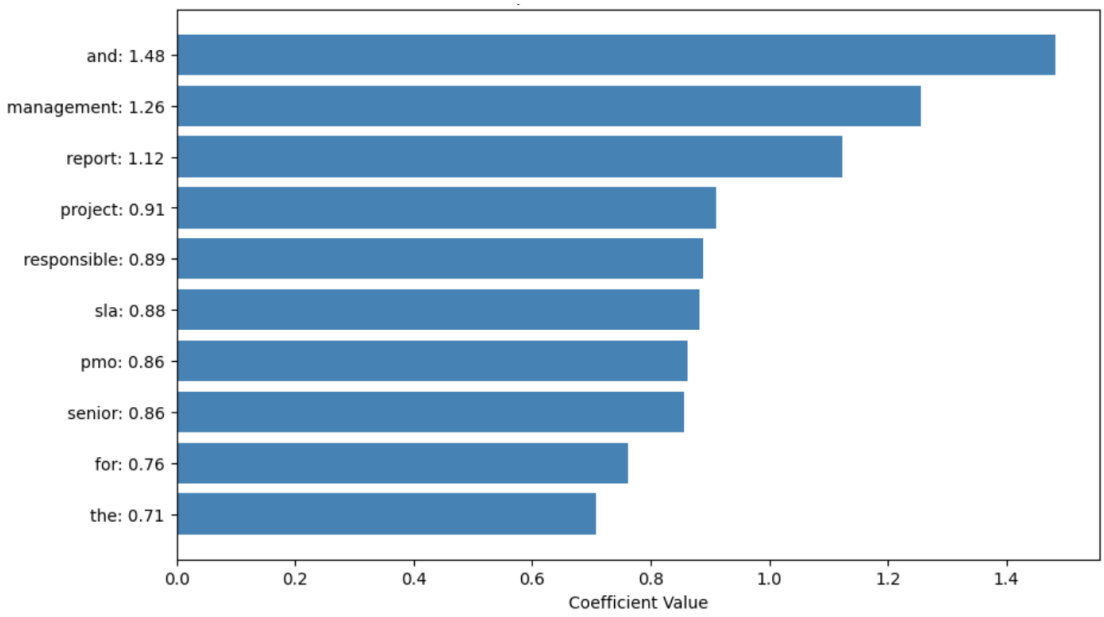
ภาพประกอบ 121 แสดงค่า *SHAP* ของ *Class 17 – Network Security Engineer*



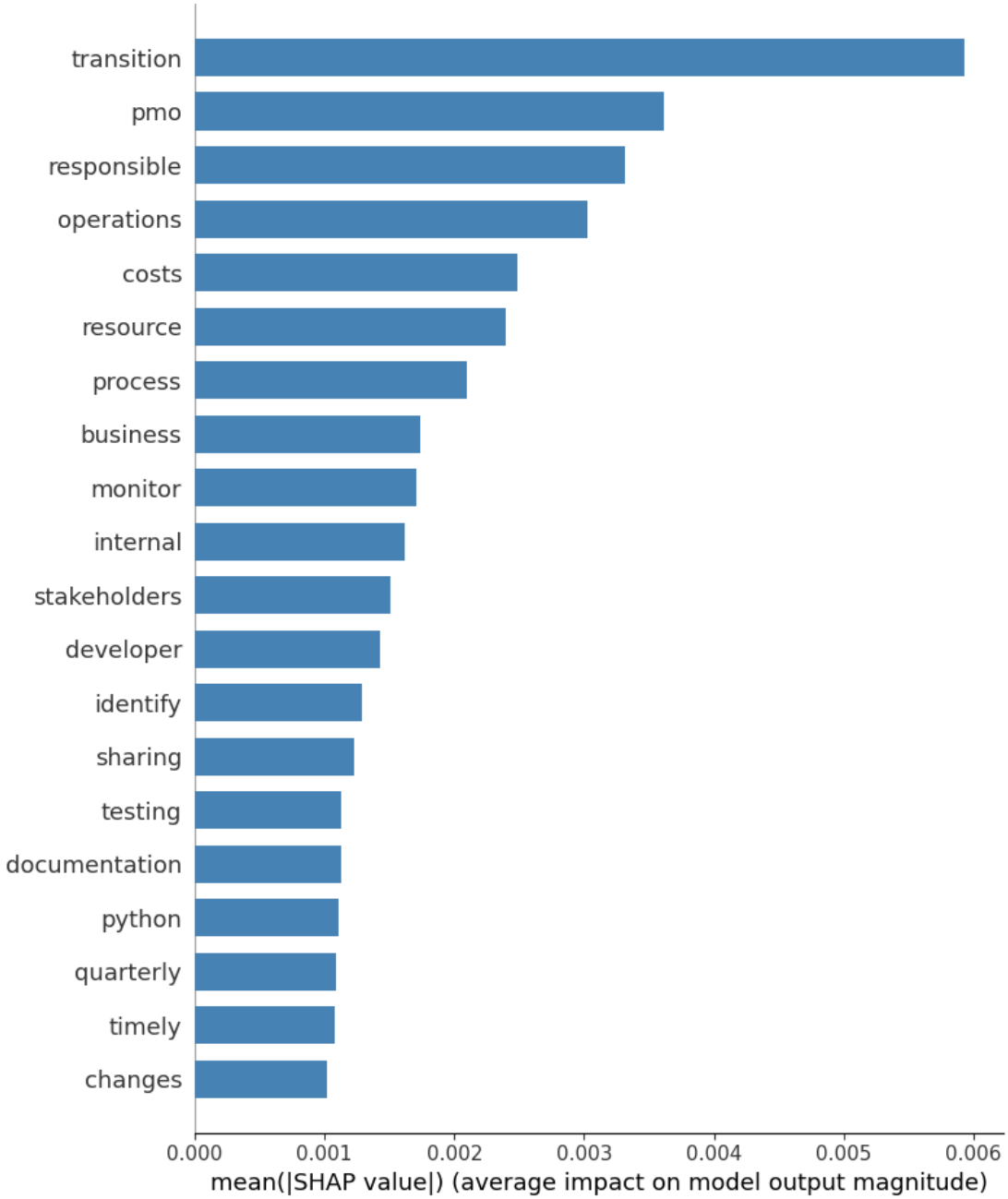
ภาพประกอบ 122 แสดงค่า *Feature Importance* ของ *Class 18 – Operations Manager*



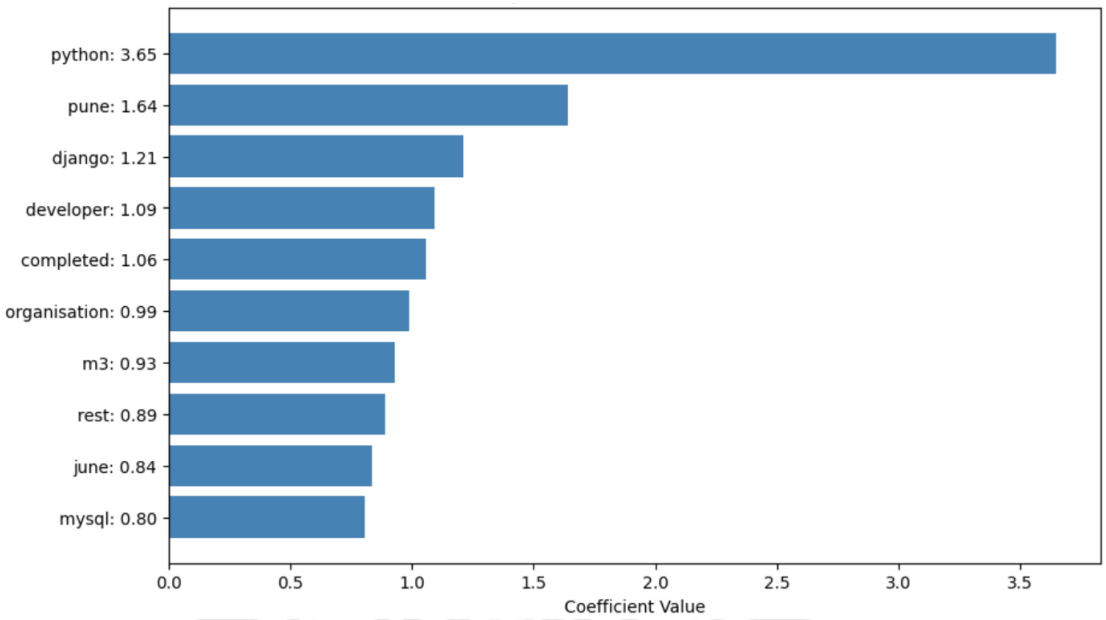
ภาพประกอบ 123 แสดงค่า *SHAP* ของ *Class 18 – Operations Manager*



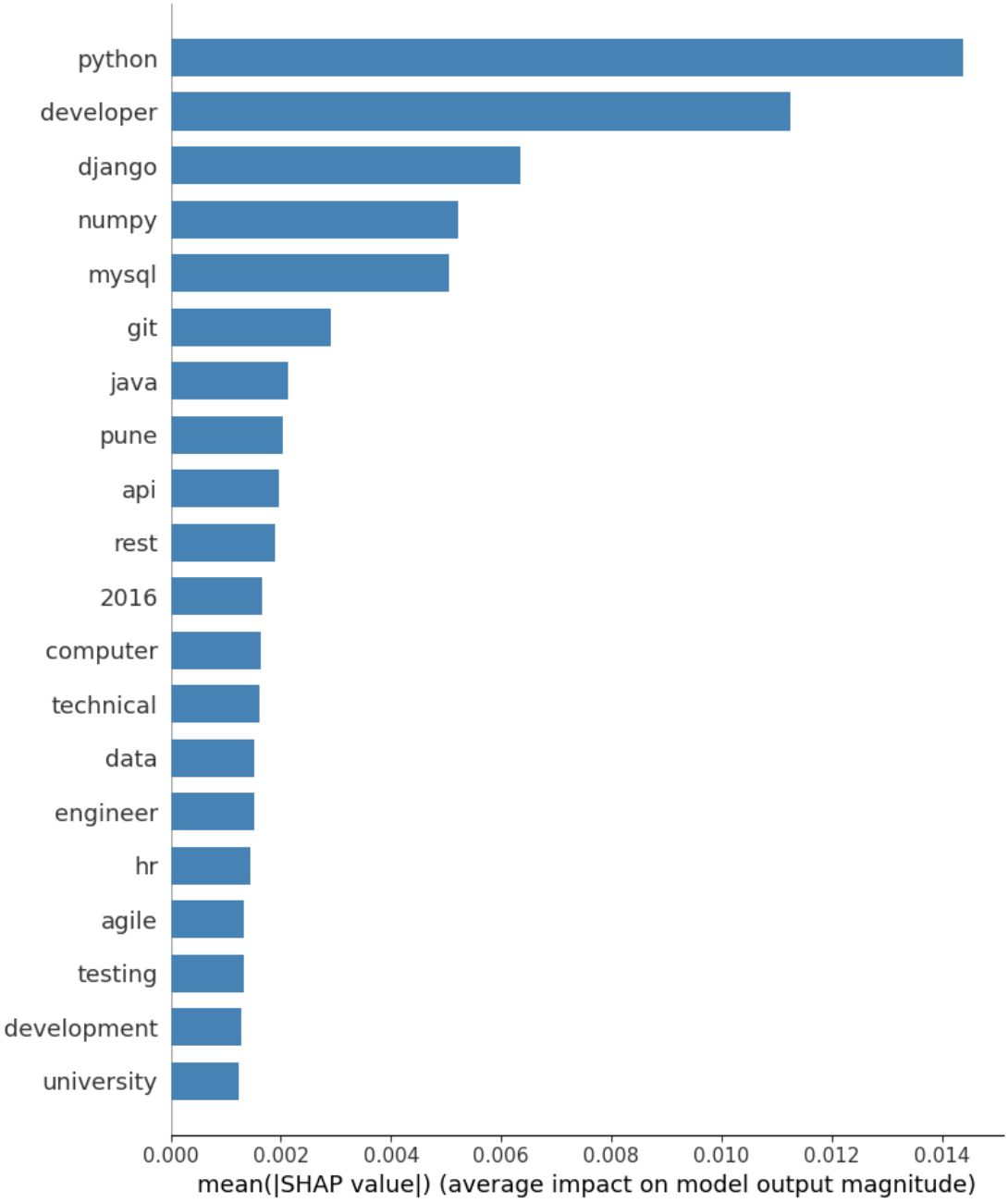
ภาพประกอบ 124 แสดงค่า *Feature Importance* ของ *Class 19 – PMO*



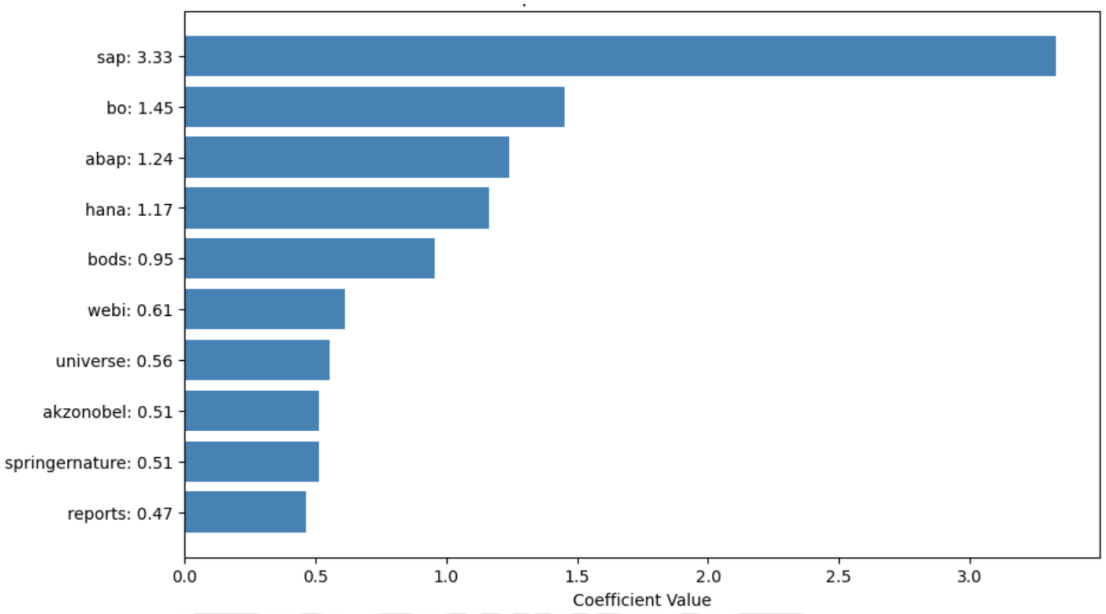
ภาพประกอบ 125 แสดงค่า *SHAP* ของ *Class 19 – PMO*



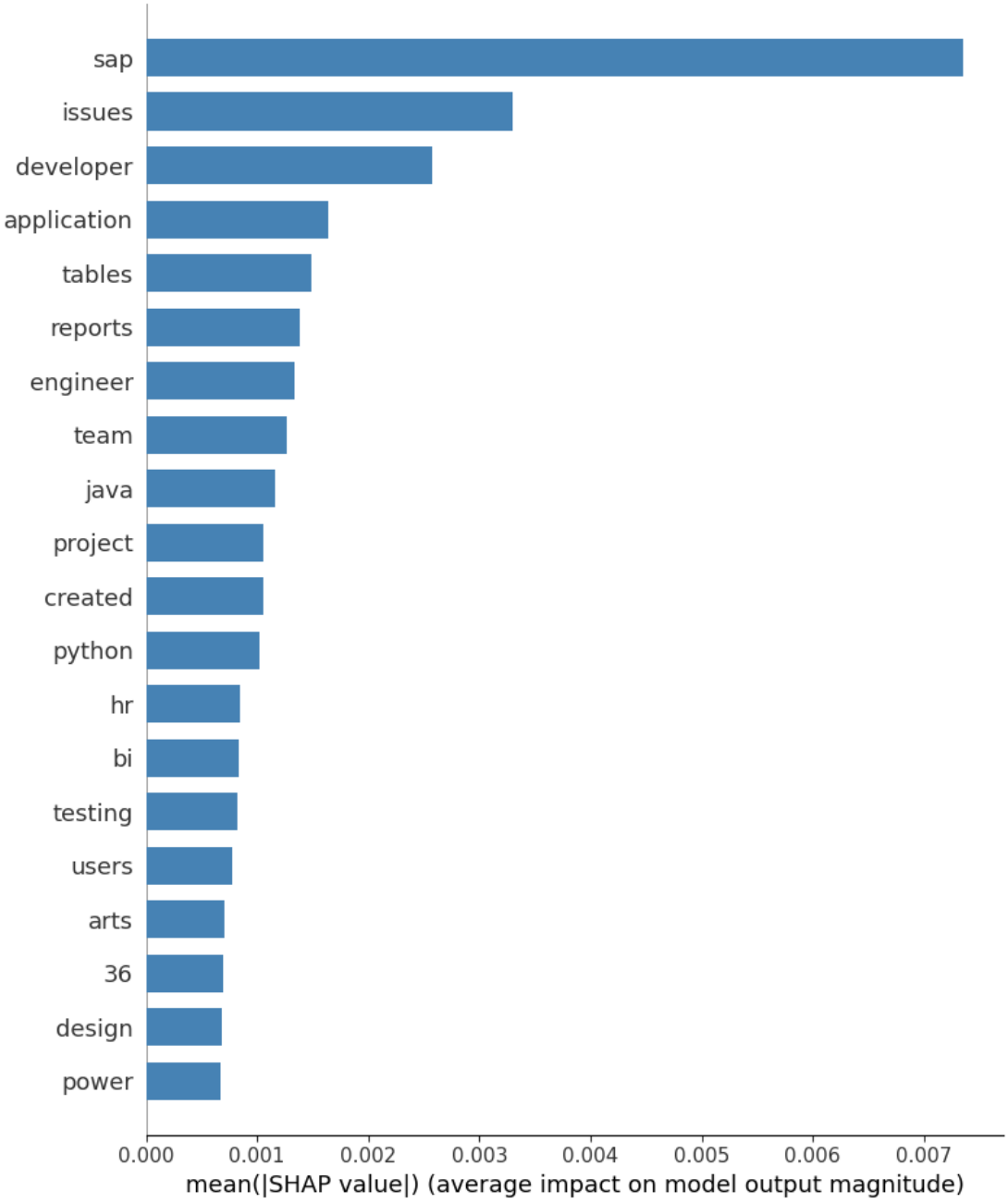
ภาพประกอบ 126 แสดงค่า *Feature Importance* ของ *Class 20 – Python Developer*



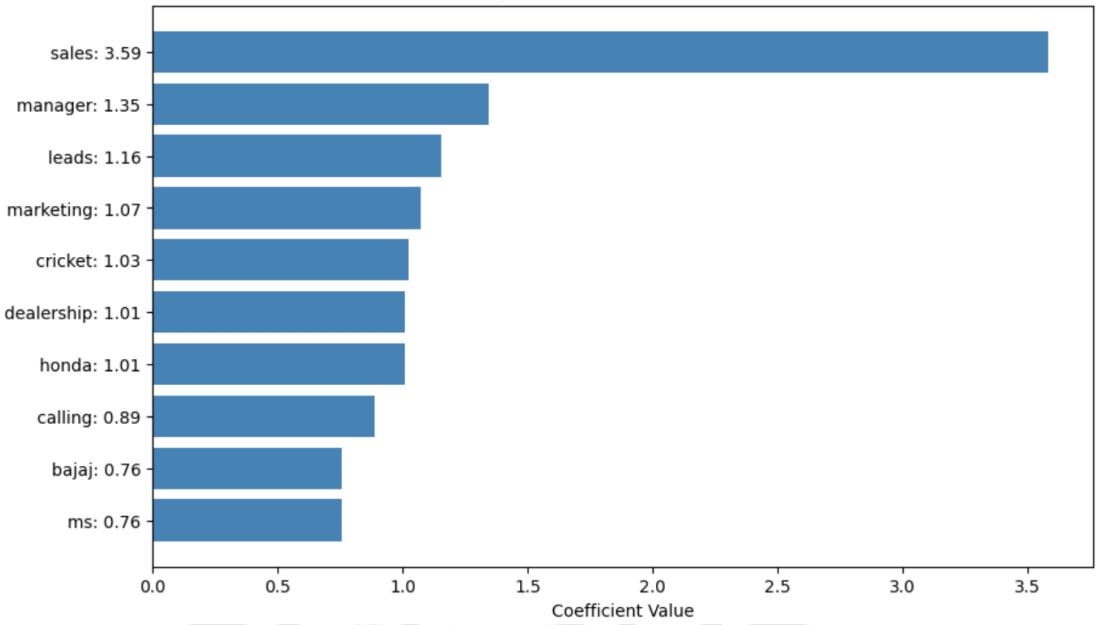
ภาพประกอบ 127แสดงค่า *SHAP* ของ *Class 20 – Python Developer*



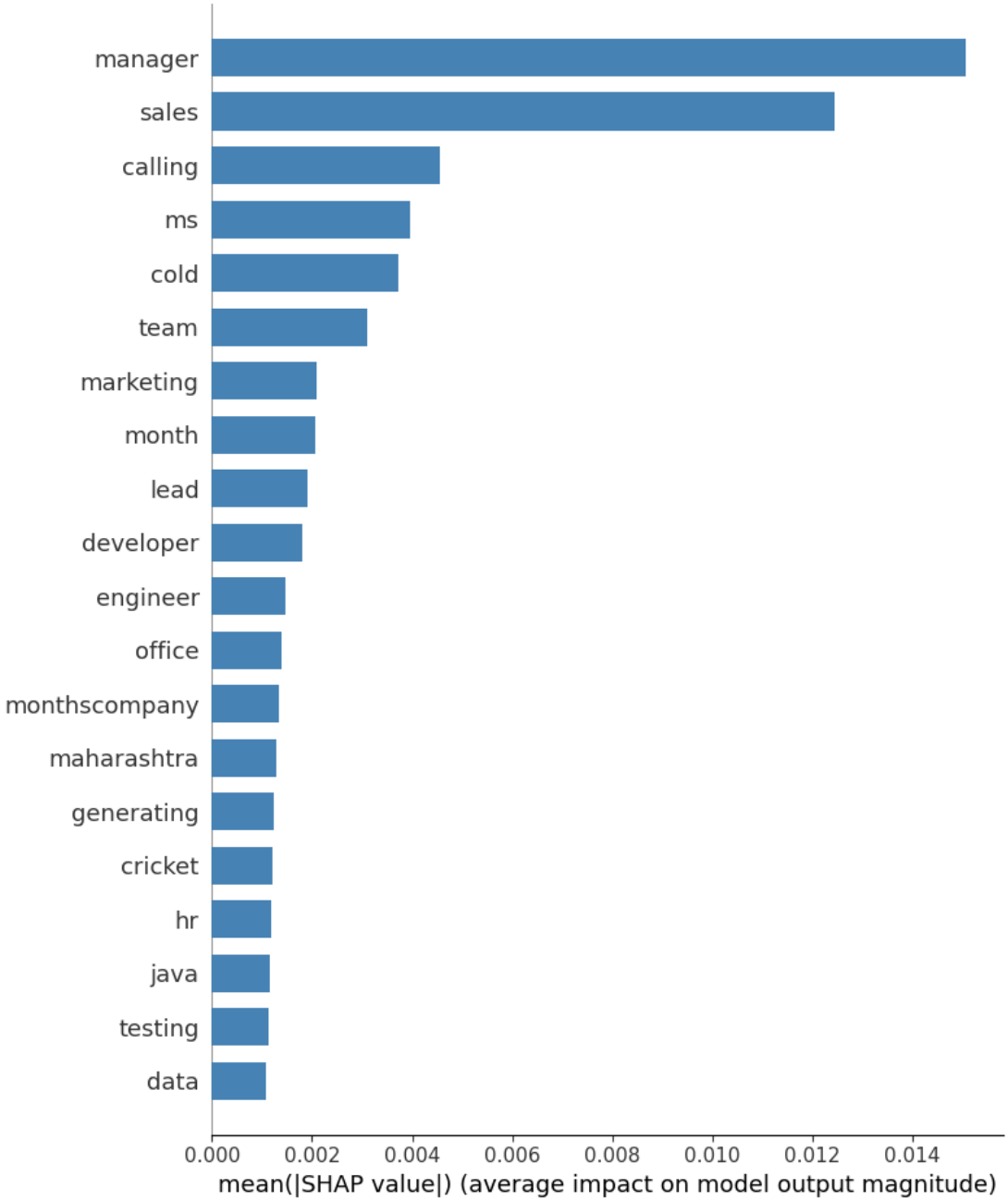
ภาพประกอบ 128 แสดงค่า *Feature Importance* ของ *Class 21 – SAP Developer*



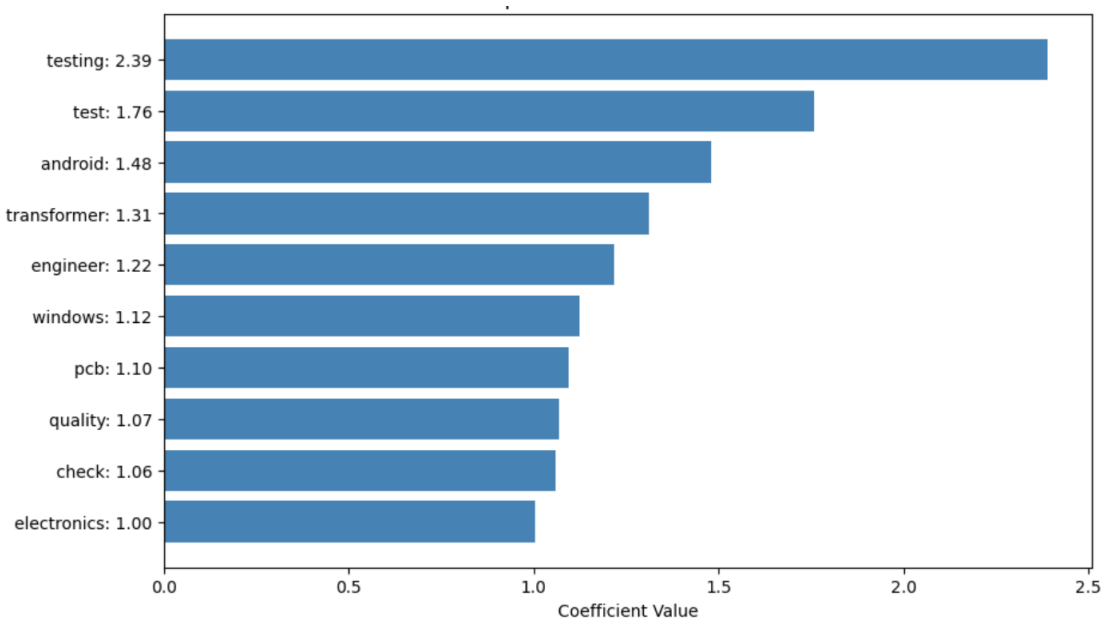
ภาพประกอบ 129 แสดงค่า *SHAP* ของ *Class 21 – SAP Developer*



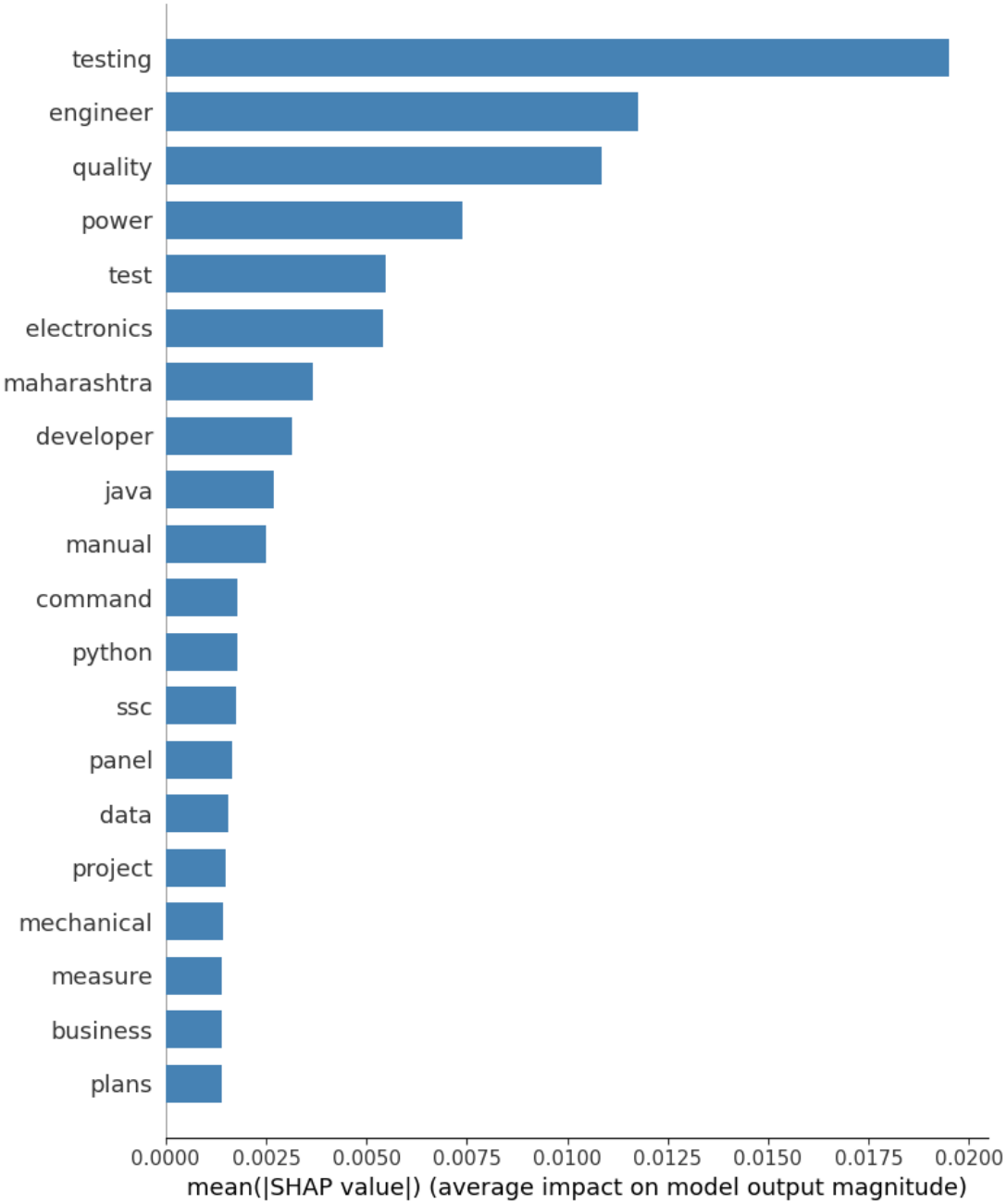
ภาพประกอบ 130 แสดงค่า *Feature Importance* ของ *Class 22 – Sales*



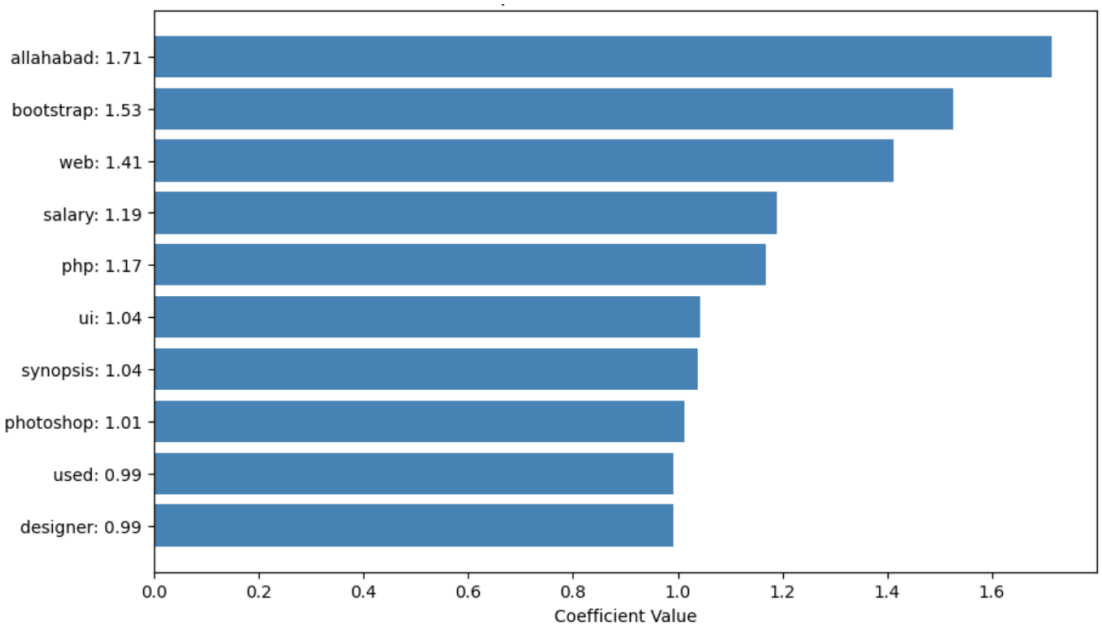
ภาพประกอบ 131 แสดงค่า *SHAP* ของ *Class 22 – Sales*



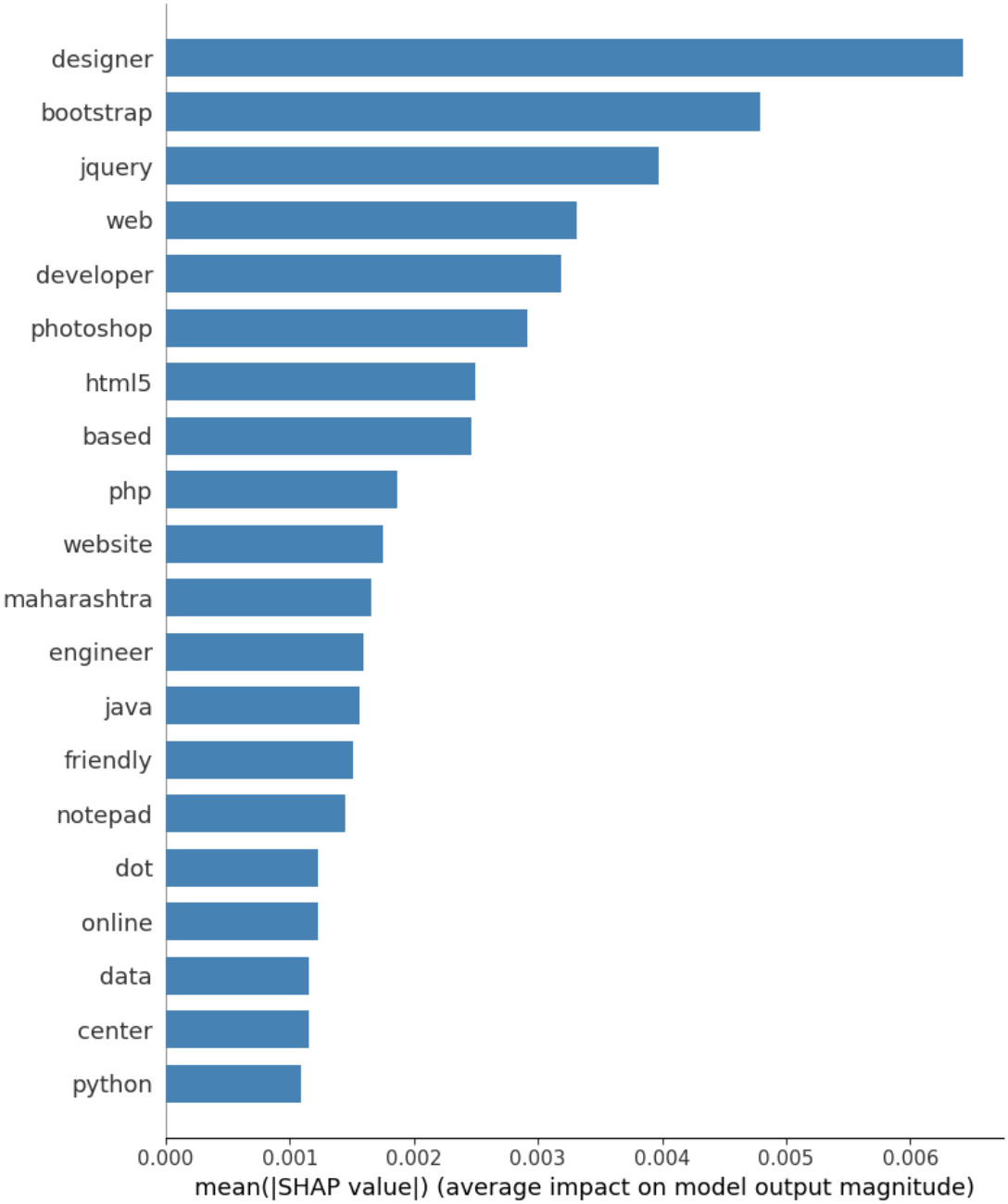
ภาพประกอบ 132 แสดงค่า *Feature Importance* ของ *Class 23 – Testing*



ภาพประกอบ 133 แสดงค่า *SHAP* ของ *Class 23 – Testing*



ภาพประกอบ 134 แสดงค่า *Feature Importance* ของ *Class 24 – Web Designing*



ภาพประกอบ 135 แสดงค่า *SHAP* ของ *Class 24 – Web Designing*

บทที่ 5
การสรุปผลการวิจัย การอภิปรายผล และข้อเสนอแนะของการวิจัย

งานวิจัยนี้มุ่งศึกษาวิธีการคัดกรองผู้สมัครจากประวัติย่อด้วยหลักการการเรียนรู้ของเครื่องและการประมวลผลภาษาธรรมชาติ โดยคัดกรองผู้สมัครจากทักษะ การศึกษา ประสบการณ์การทำงานของแต่ละสายงาน จึงได้ทำการศึกษา ทดลอง และเปรียบเทียบประสิทธิภาพของแบบจำลองการเรียนรู้ของเครื่อง และนำเสนอแบบจำลองการเรียนรู้ของเครื่องที่เหมาะสมกับการคัดกรองผู้สมัครจากประวัติย่อได้

โดยชุดข้อมูลต้นฉบับในงานวิจัยนี้มาจากเว็บไซต์ kaggle.com เป็น Public Dataset ชื่อ Updated Resume Dataset ประกอบด้วย ประเภทงาน (Category) และ ประวัติย่อ (Resume) ข้อมูล ขนาด 962 แถว 2 คอลัมน์ มีจำนวนประเภทงานที่ไม่ซ้ำกันทั้งหมด 25 ประเภทที่ผ่านการทำ labelling เรียบร้อยแล้ว

ขั้นตอนการทำงานมีทั้งหมด 7 ขั้นตอนได้แก่

- การรวบรวมข้อมูล
- การวิเคราะห์ข้อมูลเชิงสำรวจ
- การเตรียมข้อมูล
- การสร้างแบบจำลอง
- การทดสอบประสิทธิภาพของแบบจำลอง
- การประเมินแบบจำลอง
- การหาคุณลักษณะที่สำคัญ

เมื่อนำข้อมูลมาวิเคราะห์พบว่าอันดับประเภทงานที่มีประวัติย่อมากที่สุด 3 อันดับแรก ได้แก่

1. Java Developer (84 ประวัติย่อ)
2. Testing (70 ประวัติย่อ)
3. DevOps Engineer (55 ประวัติย่อ)

ซึ่งในขั้นตอนการเตรียมข้อมูล ผู้วิจัยได้นำชุดข้อมูลนี้มาเข้าสู่กระบวนการประมวลผลภาษาธรรมชาติ (NLP) ได้แก่

- ลบคำต่าง ๆ ที่ไม่สำคัญในประวัติย่อออก เช่น URLs, RT, cc, hashtags, @, ตัวอักษรพิเศษ และช่องว่าง

- นำไปแปลงข้อมูลประเภทงานจากข้อความให้เป็นข้อมูลตัวเลข โดยใช้คำสั่ง Label Encoding เพื่อให้แต่ละประเภทงานกลายเป็นคลาส

- นำเข้าสู่กระบวนการแบ่งข้อความออกเป็นหน่วยเล็ก ๆ (Tokenization) เพื่อช่วยในการทำความเข้าใจบริบทของข้อความ จากนั้น ตัด stop words หรือ คำที่เจอบ่อย ๆ แต่ไม่สื่อความหมายออก เช่น ‘were’, ‘her’, ‘more’, ‘this’ เป็นต้น สุดท้ายทำ Lemmatization หรือการเปลี่ยนรูปคำให้อยู่ในรูปแบบของคำดั้งเดิมหรือคำกริยาช่องที่ 1 เพื่อให้อยู่ในรากศัพท์เดียวกัน และสกัดคุณลักษณะที่สำคัญด้วย TF-IDF แปลงข้อมูลจาก sparse ให้เป็น dense

- ใช้ฟังก์ชัน FreqDist เพื่อดูความถี่ของคำทั้งหมดในข้อความ พบว่าคำว่า Experience มีมากที่สุด จำนวน 3,829 ครั้ง

และเข้าสู่ขั้นตอนการสร้างแบบจำลองต่าง ๆ โดยแบ่งชุดข้อมูลออกเป็นชุดข้อมูลสำหรับการฝึกแบบจำลอง (training dataset) 80% และชุดข้อมูลสำหรับการทดสอบแบบจำลอง (testing dataset) 20% ได้แก่

1. Support Vector Classification (SVC)
2. Logistic Regression, Random Forest
3. K-Nearest Neighbors (KNN)
4. Gradient Boosting, AdaBoost Classifier
5. Gaussian Naïve Bayes
6. Decision Tree

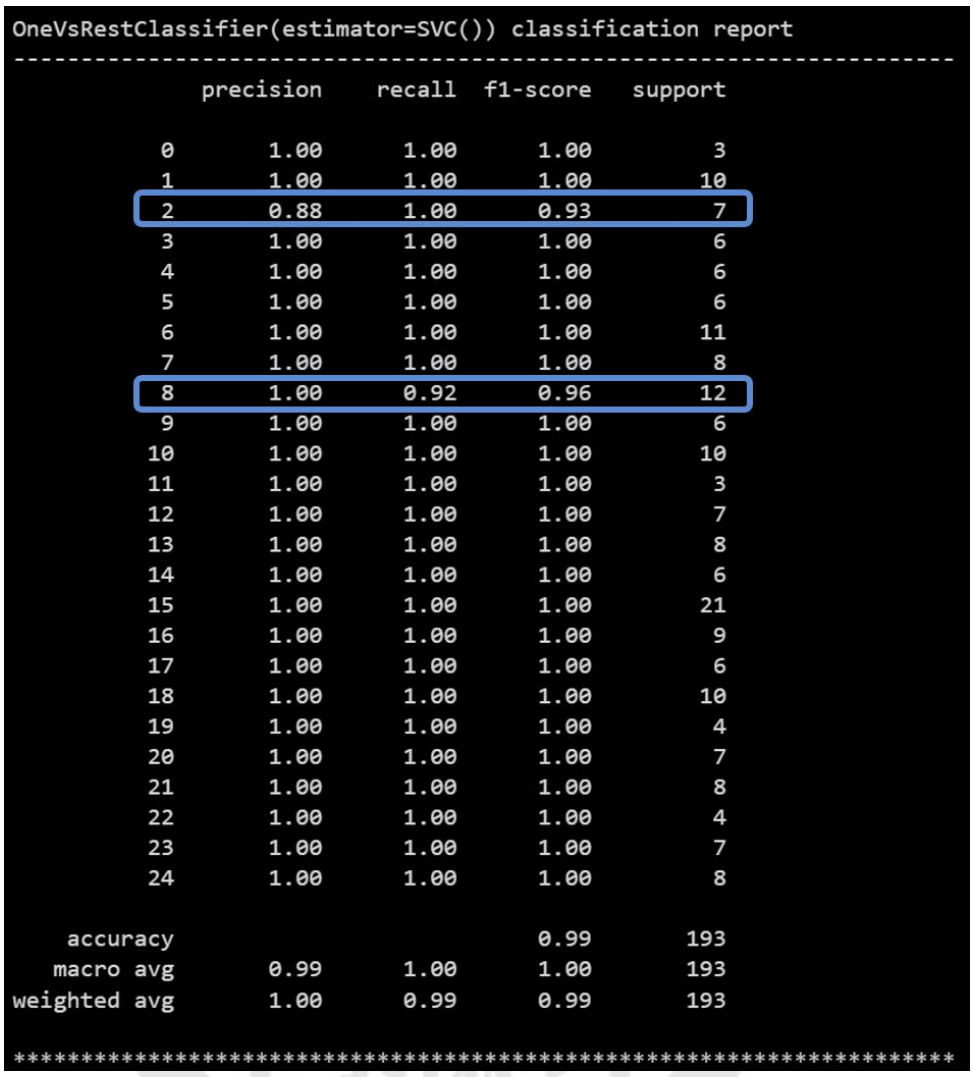
อีกทั้งยังใช้เทคนิค OneVsRestClassifier เป็นคลาสย่อยของ Classifier ในไลบรารี scikit-learn ที่ใช้เพื่อจำแนกประเภทแบบ multi-class โดยสร้างแบบจำลองย่อยแยกกันสำหรับแต่ละคลาส เช่น ในงานวิจัยนี้มีชุดข้อมูลทั้งหมด 25 คลาส ซึ่งแบบจำลอง OneVsRestClassifier จะสร้างแบบจำลองย่อย 25 แบบจำลอง แต่ละแบบจำลองจะจำแนกประเภทตัวอย่างออกเป็นคลาสเดียว

จากนั้นทำการวัดการประเมินผลของแบบจำลอง Cross Validation แบบ 5 fold เพื่อประเมินค่าความคลาดเคลื่อนของตัวชี้วัดต่าง ๆ เช่น accuracy, precision, recall, และ F1 score อีกด้วย

วิเคราะห์คุณลักษณะที่สำคัญในชุดข้อมูลของงานวิจัยนี้คือคำต่าง ๆ เพื่อทราบว่าคำใดมีผลต่อการจำแนกประเภท โดยจะใช้วิธีการหาคุณลักษณะที่สำคัญ 2 วิธีเปรียบเทียบกันคือการหาค่า Coefficient และ ค่า SHAP Values

5.1 สรุปผลการวิจัย

ผลการทดลองพบว่าแบบจำลอง Support Vector Classification (SVC), Logistic Regression และ Random Forest มีค่าความถูกต้องสูงที่สุดเท่ากับ 99.48% เมื่อวิเคราะห์ค่า Cross Validation ประกอบพบว่าแบบจำลอง Support Vector Classification (SVC) มีค่ามากที่สุดอยู่ที่ 99.50% และมีค่า Precision 99.50%, Recall 99.71%, F1 Score 99.58% เมื่อมาดูแต่ละคลาสในแบบจำลอง SVC พบว่าส่วนใหญ่ทำนายผลได้ถูกต้องแม่นยำ ซึ่งจะมีเพียง 2 คลาสเท่านั้นที่ทำนายผลได้แม่นยำน้อยลงมา ได้แก่ คลาสที่ 2 คือ Automation Testing มีค่า f1-score เท่ากับ 0.93 และคลาสที่ 8 คือ DevOps Engineer มีค่า f1-score เท่ากับ 0.96 ซึ่งผู้วิจัยสันนิษฐานว่าอาจเกิดจากข้อมูลด้านการศึกษา ทักษะ และประสบการณ์การทำงานที่มีความคล้ายคลึงกันในสองคลาสนี้ จึงทำให้การทำนายผลมีความผิดพลาดมากกว่าคลาสอื่น ๆ ดังภาพประกอบ 136



OneVsRestClassifier(estimator=SVC()) classification report

	precision	recall	f1-score	support
0	1.00	1.00	1.00	3
1	1.00	1.00	1.00	10
2	0.88	1.00	0.93	7
3	1.00	1.00	1.00	6
4	1.00	1.00	1.00	6
5	1.00	1.00	1.00	6
6	1.00	1.00	1.00	11
7	1.00	1.00	1.00	8
8	1.00	0.92	0.96	12
9	1.00	1.00	1.00	6
10	1.00	1.00	1.00	10
11	1.00	1.00	1.00	3
12	1.00	1.00	1.00	7
13	1.00	1.00	1.00	8
14	1.00	1.00	1.00	6
15	1.00	1.00	1.00	21
16	1.00	1.00	1.00	9
17	1.00	1.00	1.00	6
18	1.00	1.00	1.00	10
19	1.00	1.00	1.00	4
20	1.00	1.00	1.00	7
21	1.00	1.00	1.00	8
22	1.00	1.00	1.00	4
23	1.00	1.00	1.00	7
24	1.00	1.00	1.00	8
accuracy			0.99	193
macro avg	0.99	1.00	1.00	193
weighted avg	1.00	0.99	0.99	193

****

ภาพประกอบ 136 แสดงคลาสที่มีค่า *f1-score* น้อยของแบบจำลอง *SVC*

อีกทั้งยังได้ทำการหาคุณลักษณะที่มีความสำคัญต่อการจำแนกประเภทของสายงานจากประวัติย่อ โดยใช้วิธีการคำนวณค่า Coefficients และค่า SHAP Value ซึ่งคุณลักษณะเหล่านี้ จะช่วยให้เข้าใจถึงปัจจัยที่ส่งผลต่อการจำแนกประเภทสายงานได้ดียิ่งขึ้น และสามารถนำไปปรับปรุงและพัฒนาโมเดลต่อไปในอนาคต

กล่าวโดยสรุปคือแบบจำลองที่มีประสิทธิภาพดีที่สุดและเหมาะสมที่สุดในการคัดกรองผู้สมัครจากประวัติย่อได้แก่แบบจำลอง SVC

5.2 ข้อจำกัดงานวิจัย

5.2.1 แบบจำลองที่สร้างขึ้นรองรับเฉพาะข้อความภาษาอังกฤษเท่านั้น เนื่องจากภาษาที่แตกต่างกันมีวิธีการจัดการข้อมูลหรือการตัดแบ่งคำที่ไม่เหมือนกัน

5.2.2. ชุดข้อมูลเรซูเม่ของแต่ละประเภทงานนำมาจากเว็บไซต์ Kaggle เพียงแห่งเดียวเท่านั้น ซึ่งส่วนใหญ่เป็นสายงานด้านไอที อาจไม่ครอบคลุมในสายงานอื่น ๆ

5.2.3 ข้อมูลที่ใช้ในงานวิจัยถูกติดป้ายชื่อ (labeled data) ประเภทสายงานไว้แล้ว อาจส่งผลให้ประสิทธิภาพการจำแนกประเภทสายงานคลาดเคลื่อนจากความเป็นจริงได้

5.3 ข้อเสนอแนะ

5.3.1 เพิ่มข้อมูลเรซูเม่จากแหล่งอื่น ๆ ที่มีความหลากหลายมากขึ้น

5.3.2 ปรับจูนโมเดลต่าง ๆ ให้มีประสิทธิภาพสูงขึ้น

5.3.3 เปรียบเทียบประสิทธิภาพวิธีการสกัดคุณลักษณะสำคัญด้วย TF-IDF กับวิธีอื่น เช่น Word2Vec เป็นต้น

5.3.4 เปรียบเทียบประสิทธิภาพของ Feature Importance จาก 1 คำ เป็น 2,3 คำ

5.3.5 ทดลองและเปรียบเทียบประสิทธิภาพของแบบจำลอง Deep Learning, Transformer เพิ่มเติม เช่น LSTM, CNN เป็นต้น

บรรณานุกรม

Achieve.Plus. (2563, 4 สิงหาคม). *4 Classification* ที่สำคัญใน *Supervised Learning*. Medium. https://medium.com/achieve-space/4-classification-ที่สำคัญใน-supervised-learning-a64e75250141

Ali, I., Mughal, N., Khan, Z. H., Ahmed, J., & Mujtaba, G. (2022). Resume Classification System using Natural Language Processing and Machine Learning Techniques. *Mehran University Research Journal of Engineering and Technology, 41*(1), 65-79. https://doi.org/10.22581/muet1982.2201.07

Anupoomchaiya, P. (2021). *Venous Thromboembolism Diagnosis Based On Machine Learning* [Unpublished master's thesis]. Srinakharinwirot University.

Bhavsar, K. (2023, January 6). *Stemming: Porter Vs. Snowball Vs. Lancaster*. Towards AI. https://towardsai.net/p/l/stemming-porter-vs-snowball-vs-lancaster

Brownlee, J. (2020, August 19). *4 Types of Classification Tasks in Machine Learning*. Machine Learning Mastery. https://machinelearningmastery.com/types-of-classification-in-machine-learning/

Daroontham, W. (2561a, 21 พฤศจิกายน). เจาะลึก *Random Forest !!!— Part 2 of "รู้จัก Decision Tree, Random Forest,* และ *XGBoost!!!"*. Medium. https://medium.com/@witchapongdaroontham/เจาะลึก-random-forest-part-2-of-รู้จัก-decision-tree-random-forest-และ-xgboost-79b9f41a1c1c

Daroontham, W. (2561b, 31 ตุลาคม). ขั้นตอนการเตรียมข้อมูลประเภท *Text* ภาษาไทย แบบง่ายๆ โดยใช้ *Python (Simple Thai text preprocessing using Python)*. Medium. https://medium.com/@witchapongdaroontham/ขั้นตอนการเตรียมข้อมูลประเภท-text-ภาษาไทย-แบบง่ายๆ-โดยใช้-python-simple-thai-text-preprocessing-c8c46ca3ce46

datawow. (2563, 17 มิถุนายน). ขอให้โชคดีมีชัยในโลก *NLP — Part 1*. Data Wow. https://www.datawow.io/blogs/wish-you-luck-in-nlp-world

Dong, Z. (2023). Resume recommendation based on text similarity. *Proceedings of the 3rd International Conference on Signal Processing and Machine Learning, 6*(1), 848-853. https://doi.org/10.54254/2755-2721/6/20230937

Dutta, G. (2021). *Resume Dataset*. kaggle. https://www.kaggle.com/datasets/gauravduttakiit/resume-dataset

HREX.asia. (2562, 13 มีนาคม). การสรรหาบุคลากร (*Recruitment)* เพื่อเข้าร่วมงานกับองค์กร. https://th.hrnote.asia/recruit/190313-recruitment/

Kanoktipsatharporn, S. (2566, 23 พฤศจิกายน). *Natural Language Processing (NLP)* คืออะไร รวมคำศัพท์เกี่ยวกับ *Natural Language Processing (NLP) – NLP ep.1*. BUA Labs. https://www.bualabs.com/archives/119/what-is-nlp-natural-language-processing-nlp-task-in-thai-nlp-ep-1/

Kullawattana, T. (2562, 17 ตุลาคม). *Introduction NLP with spaCy in Software Engineering Part 3 (Word Stemming* และ *Lemmatization)*. Medium. https://suttipong-kull.medium.com/word-stemming-และ-lemmatization-5bd99ff3af96

Majumder, P. (2022, August 26). *Text Analytics of Resume Dataset with NLP!* Analytics Vidhya. https://www.analyticsvidhya.com/blog/2021/06/text-analytics-of-resume-dataset-with-nlp/

Mankawade, A., Pungliya, V., Bhonsle, R., Pate, S., Purohit, A., & Raut, A. (2023, April 7-9). *Resume Analysis and Job Recommendation* 2023 IEEE 8th International Conference for Convergence in Technology (I2CT), India.

Mr.P L. (2561, 15 พฤศจิกายน). *SVM* อดีตเคยหวานปัจจุบันแอบเซง : *Machine Learning 101*. Medium. https://medium.com/mmp-li/svm-อดีตเคยหวานปัจจุบันแอบเซง-machine-learning-101-6008753c780c

Pal, R., Shaikh, S., Satpute, S., & Bhagwat, S. (2022, 01/01). Resume Classification using various Machine Learning Algorithms. *ITM Web of Conferences, 44*, 03011. https://doi.org/10.1051/itmconf/20224403011

Paparrizos, I., Cambazoglu, B., & Gionis, A. (2011). Machine learned job recommendation. *Proceedings of the 2011 ACM Conference on Recommender Systems, RecSys 2011*, 325-328. https://doi.org/10.1145/2043932.2043994

Prasertsom, P. (2563, 1 ตุลาคม). สกัดใจความสำคัญของข้อความด้วยเทคนิคการประมวลผลทางภาษาเบื้องต้น: *TF-IDF, Part 1*. Big Data Institute. https://bdi.or.th/big-data-101/tf-idf-1/

Premanand, S. (2566, 16 พฤศจิกายน). *The A-Z guide to Support Vector Machine*. Analytics Vidhya. https://www.analyticsvidhya.com/blog/2021/06/support-vector-machine-better-understanding/

Roy, P., Chowdhary, S., & Bhatia, R. (2020). A Machine Learning approach for automation of Resume Recommendation system. *Procedia Computer Science, 167*, 2318-2327. https://doi.org/10.1016/j.procs.2020.03.284

Sayfullina, L., Malmi, E., Liao, Y., & Jung, A. (2017, December 21). Domain Adaptation for Resume Classification Using Convolutional Neural Networks. *Analysis of Images, Social Networks and Texts*, 82-93. https://doi.org/https://doi.org/10.1007/978-3-319-73013-4_8

scikit-learn. (2023). *3.1. Cross-validation: evaluating estimator performance*. https://scikit-learn.org/stable/modules/cross_validation.html

Sharma, M., Choudhary, G., & Susan, S. (2023). Resume Classification using Elite Bag-of-Words Approach. *Proceedings of the 5th International Conference on Smart Systems and Inventive Technology (ICSSIT 2023)*, 1409-1413. https://doi.org/10.1109/ICSSIT55814.2023.10061036

spaCy. (2023, September 6). *Part-of-speech tagging*. https://spacy.io/usage/linguistic-features#pos-tagging

Sruthi, P., Adithya, P. N. V. K. G., Suleman, M. D., Kunal, P., & Gairola, S. P. (2023). Smart Resume Analyser: A Case Study using RNN-based Keyword Extraction. *E3S Web of Conferences, 430*, 1-11. https://doi.org/10.1051/e3sconf/202343001023

Suhas, H. E., & Manjunath, A. E. (2020). Differential Hiring using a Combination of NER and Word Embedding. *International Journal of Recent Technology and Engineering (IJRTE), 9*(1), 1344-1349. https://doi.org/10.35940/ijrte.A2400.059120

Zubeda, J. A. A., Shaheen, M. A. A., Godavari, G. R. N., & Naseem, S. Z. M. S. (2015). *Resume Ranking using NLP andMachine Learning*. studocu. https://www.studocu.com/in/document/marwadi-university/pc-major-project/major-reference-i-hope-this-will-be-helpful-for-your-project/44631121

ชิตพงษ์ กิตตินราดร. (2563). *Support Vector Machines*. GitHub. https://guopai.github.io/ml-blog08.html

วิกิพีเดีย. (2566, 1 กรกฎาคม). โครงข่ายประสาทเทียม. https://th.wikipedia.org/wiki/โครงข่ายประสาทเทียม

สำนักงานสถิติแห่งชาติ. (2566). ประเภทของข้อมูลสถิติ. http://service.nso.go.th/nso/knowledge/estat/esta1_3.html

ประวัติผู้เขียน