การคัดกรองผู้สมัครจากประวัติย่อด้วยหลักการเรียนรู้ของเครื่อง

นงนุช เสถียรกิตติโรจน์[1*], วีระ สอิ้ง[2]

บทคัดย่อ

บุคลากรถือเป็นทรัพยากรที่มีความสำคัญมากที่สุดขององค์กร ดังนั้นการสรรหาบุคลากรที่มีทักษะความสามารถเหมาะสมกับตำแหน่งงานจึงมีความสำคัญอย่างยิ่ง ในปัจจุบันผู้คัดสรรจะทำการพิจารณาผู้สมัครงานด้วยวิธีการอ่านข้อมูลจากในประวัติย่อด้วยสายตาของมนุษย์เป็นหลัก และด้วยความที่มีใบสมัครเยอะ ทำให้เกิดความผิดพลาด และความล่าช้าขึ้น เพื่อเพิ่มประสิทธิภาพและความรวดเร็วของการคัดสรรจึงทำให้เกิดงานวิจัยนี้ขึ้น วัตถุประสงค์เพื่อศึกษาสร้างแบบจำลองการคัดกรองผู้สมัครจากประวัติย่อด้วยการเรียนรู้ของเครื่องประเภทการเรียนรู้แบบมีผู้สอน กระบวนการทำงานของการประมวลผลภาษาธรรมชาติ ทดลอง และเปรียบเทียบประสิทธิภาพของแบบจำลอง 8 แบบ ได้แก่ Support Vector Classification (SVC), Logistic Regression, Random Forest, K-Nearest Neighbors (KNN), Gradient Boosting, AdaBoost Classifier, Gaussian Naïve Bayes, Decision Tree โดยใช้ข้อมูลจากแหล่งข้อมูลสาธารณะ [2] ซึ่งในชุดข้อมูลประกอบด้วยประเภทงานและประวัติย่อ มีขนาด 962 แถว 2 คอลัมน์ จากผลการทดลองพบว่าแบบจำลอง Support Vector Classification มีประสิทธิภาพดีที่สุดเมื่อเทียบกับแบบจำลองอื่น ๆ ได้ค่าความถูกต้อง (Accuracy score) อยู่ที่ 99.4% ค่า Cross Validation อยู่ที่ 99.5% อีกทั้งศึกษาคำที่มีคุณลักษณะสำคัญ (Feature Important) ในการจำแนกประเภทสายงานอีกด้วย ได้แก่คำว่า “developer” มีค่า Feature important มากที่สุดอยู่ที่ 0.0119 จากผลลัพธ์เหล่านี้ แบบจำลอง SVC ที่เสนอจะช่วยให้ผู้คัดเลือกบุคลากรเลือกผู้สมัครที่เหมาะสมกับงานได้อย่างมีประสิทธิภาพและแม่นยำ

คำสำคัญ : การคัดกรองประวัติย่อ, การเรียนรู้ของเครื่อง, การประมวลผลภาษาธรรมชาติ, การจำแนกประเภทหลายหมวดหมู่

---

[1] หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิทยาการข้อมูล คณะวิทยาศาสตร์ มหาวิทยาลัยศรีนครินทรวิโรฒ กรุงเทพฯ 10110

[2] คณะวิทยาศาสตร์ มหาวิทยาลัยศรีนครินทรวิโรฒ กรุงเทพฯ 10110

* Corresponding author: Tel.: 088-6116662 E-mail address: nongnuch.bee@g.swu.ac.th

Candidate Screening from Resume Using Machine Learning

Nongnuch Satheankittiroj[1*], Vera Sa-ing[2]

Abstract

Personnel are considered the most valuables resource of an organization. Therefore, selecting suitable personnel who have skills and abilities for the position are important. Currently, recruiters typically review job applicants by reading the resumes that can lead to errors and delays especially when faced with large number of applications. To enhance efficiency and speed up the selection process, this research aims to study and create a machine learning model for screening the job applicants from the main topics filling in the resumes by using supervised machine learning and natural language processing techniques. This research experimented and compared the performance of 8 models that consist of Support Vector Classification (SVC), Logistic Regression, Random Forest, K-Nearest Neighbors (KNN), Gradient Boosting, AdaBoost Classifier, Gaussian Naive Bayes, and Decision Tree by using dataset from public sources. This dataset consists of job types and resumes that have a total of 962 rows and 2 columns. From the experimental results, this research proposed the trained SVC model that represents the best model from an accuracy score of 99.4% and a cross-validation score of 99.5%. In addition, our study identifies important features for job classification, such as the term "developer" having the highest feature importance score of 0.0119. From these results, the proposed SVC model will suggest the recruiters to select the suitable candidate efficiently and accurately with suitable job.

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

[1] Data Science, Faculty of Science, Srinakharinwirot University, Bangkok, 10110, Thailand

[2] Faculty of Science, Srinakharinwirot University, Bangkok, 10110, Thailand

* Corresponding author: Tel.: 088-6116662 E-mail address: nongnuch.bee@g.swu.ac.th

บทนำ

1. ที่มาและความสำคัญของงานวิจัย

การสรรหาบุคลากรนั้นมีความสำคัญต่อองค์กรเป็นอย่างมาก เนื่องจากบุคลากรถือเป็นทรัพยากรที่มีความสำคัญมากที่สุดขององค์กร ซึ่งบุคลากรไม่ว่าจะตำแหน่งใดก็ตามเปรียบเสมือนฟันเฟืองที่มีความสำคัญไม่แพ้กัน ถ้าฟันเฟืองหรือตำแหน่งใดตำแหน่งหนึ่งขาดไปอาจทำให้องค์กรประสบปัญหาไม่สามารถบรรลุเป้าหมายและพบกับความสำเร็จได้ ในทางตรงกันข้าม ถ้าหากองค์กรใดมีบุคลากรที่ทำงานได้อย่างมีประสิทธิภาพครบทุกตำแหน่ง องค์กรนั้นจะก้าวไปสู่ความสำเร็จได้อย่างไม่ยาก ดังนั้นก่อนที่จะมีบุคลากรที่มีความสามารถอยู่ในองค์กรได้นั้น ต้องอาศัยกระบวนการสรรหาบุคลากรที่มีประสิทธิภาพ

กระบวนการสรรหาบุคลากรโดยทั่วไปสามารถแบ่งออกเป็น 7 ขั้นตอน ดังนี้ [1]

1. วางแผนการสรรหาและคัดเลือก
2. กำหนดรายละเอียดของงานและคุณสมบัติของผู้สมัคร
3. สื่อสารการรับสมัครงาน
4. การคัดสรรบุคลากร
    - คัดกรองด้วยสายตามนุษย์
    - คัดกรองแบบอัตโนมัติด้วยหลักการเรียนรู้ของเครื่อง (Machine Learning)
5. การสัมภาษณ์งาน
6. การเซ็นสัญญาจ้างงาน
7. ประเมินผลการสรรหา

กระบวนการสรรหาบุคลากรนั้นใช้เวลาค่อนข้างมาก โดยเฉพาะขั้นตอนการคัดกรองผู้สมัครงานจากประวัติย่อ (Resume) ที่ผู้คัดสรรจะต้องใช้สายตาอ่านใบสมัคร และคัดเลือกผู้ที่มีความสามารถตรงกับที่องค์กรต้องการมากที่สุดในระยะเวลาที่จำกัด นั่นจึงเป็นความท้าทายอย่างมากของผู้คัดสรร ยิ่งไปกว่านั้นถ้าผู้คัดสรรอยู่ในองค์กรที่มีชื่อเสียงและได้รับความนิยมจากผู้สมัครงานมาก อาจมีใบสมัครเข้ามากว่า 1,000 ใบสมัครต่อหนึ่งวันต่อหนึ่งตำแหน่งงานก็เป็นได้ ถ้าใช้สายตามนุษย์ในการคัดกรองอย่างเดียวนั้นอาจเกิดความผิดพลาดได้ง่าย (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

2. วัตถุประสงค์ของงานวิจัย

2.1 เพื่อศึกษากระบวนการคัดกรองผู้สมัครจากประวัติย่อแบบดั้งเดิม

2.2 เพื่อศึกษาสร้างแบบจำลองการคัดกรองผู้สมัครจากประวัติย่อด้วยการเรียนรู้ของเครื่อง (Machine Learning) ประเภทการเรียนรู้แบบมีผู้สอน (Supervised Machine Learning)

2.3 เพื่อศึกษากระบวนการทำงานของการประมวลผลภาษาธรรมชาติ (Natural Language Processing)

2.4 เพื่อศึกษา ทดลอง และเปรียบเทียบประสิทธิภาพ ของแบบจำลองการเรียนรู้ของเครื่อง (Machine Learning)

2.5 เพื่อนำเสนอแบบจำลองการเรียนรู้ของเครื่องที่เหมาะสมกับการคัดกรองผู้สมัครจากประวัติย่อ

3. ขอบเขตของงานวิจัย

3.1 ข้อมูลที่ใช้ในงานวิจัยฉบับนี้มาจาก Public Dataset ชื่อ Updated Resume Dataset

3.2 ทดลองการคัดเลือกข้อมูลจากใบสมัครด้วยวิธีการจำแนกประเภทจากการเรียนรู้ของเครื่อง (Classification from Supervised Machine Learning)

3.3 ประเมินผลของแบบจำลองการเรียนรู้ของเครื่อง (Machine Learning) ด้วยการประเมินค่าแบบเชิงปริมาณ (Quantitative measurements)

4. ประโยชน์ที่คาดว่าจะได้รับจากงานวิจัย

4.1 ได้รับความรู้ความเข้าใจเกี่ยวกับกระบวนการคัดกรองผู้สมัครจากประวัติย่อแบบดั้งเดิม

4.2 ด้รับความรู้ความเข้าใจเกี่ยวกับการสร้างแบบจำลองการคัดกรองผู้สมัครจากประวัติย่อด้วยการเรียนรู้ของเครื่อง (Machine Learning)

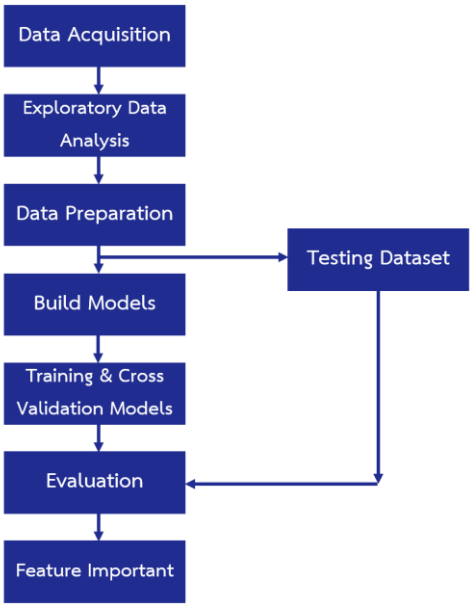
4.3 ได้รับความรู้ความเข้าใจเกี่ยวกับกระบวนการทำงานของการประมวลผลภาษาธรรมชาติ (Natural Language Processing)

4.4 สามารถทดลองและเปรียบเทียบประสิทธิภาพของแบบจำลองการเรียนรู้ของเครื่อง (Machine Learning) ในแต่ละแบบจำลองได้

4.5 สามารถนำเสนอแบบจำลองการเรียนรู้ของเครื่อง (Machine Learning) ที่เหมาะสมกับการคัดกรองผู้สมัครจากประวัติย่อได้

วิธีดำเนินการ

ขั้นตอนการทำงานวิจัย ดังภาพที่ 1



ภาพที่ 1 ขั้นตอนการทำงานวิจัย

ขั้นตอนที่ 1 : การรวบรวมข้อมูล (Data Acquisition)

เนื่องจากข้อมูลประวัติย่อ เป็นข้อมูลที่มีความเป็นส่วนตัว และละเอียดอ่อน จึงทำให้ยากต่อการหาชุดข้อมูลจริงของผู้สมัคร ดังนั้นผู้วิจัยจึงใช้ข้อมูลจาก kaggle.com ซึ่งเป็น Public Dataset ชื่อ Updated Resume Dataset [2] มาใช้ในงานวิจัยนี้ ซึ่งในชุดข้อมูลประกอบด้วย ประเภทงาน (Category) และ ประวัติย่อ (Resume)

ตาราง 1 ตัวอย่างชุดข้อมูล

```
1 df = pd.read_csv("/content/drive/MyDrive/Bee Nongnuch.IS/UpdatedResumeDataSet.csv")
2 df.head(10)
```

	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...
5	Data Science	SKILLS C Basics, IOT, Python, MATLAB, Data Sci...
6	Data Science	Skills â□¢ Python â□¢ Tableau â□¢ Data Visuali...
7	Data Science	Education Details \r\n B.Tech Rayat and Bahr...
8	Data Science	Personal Skills â□¢ Ability to quickly grasp t...
9	Data Science	Expertise â□□ Data and Quantitative Analysis â...

ขั้นตอนที่ 2 : การวิเคราะห์ข้อมูลเชิงสำรวจ (Exploratory Data Analysis: EDA)

ขั้นตอนต่อไปคือการทำความเข้าใจกับข้อมูล ซึ่งพบว่าข้อมูลมีขนาด 962 แถว 2 คอลัมน์ ชนิดข้อมูลแบบ objective และไม่มีค่าว่าง (Missing Value) มีจำนวนประเภทงานที่ไม่ซ้ำกันทั้งหมด 25 ประเภท โดยแสดงแผนภูมิวงกลมแสดงจำนวนประวัติย่อของแต่ละประเภทงาน ดังภาพที่ 4

```
 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())
```

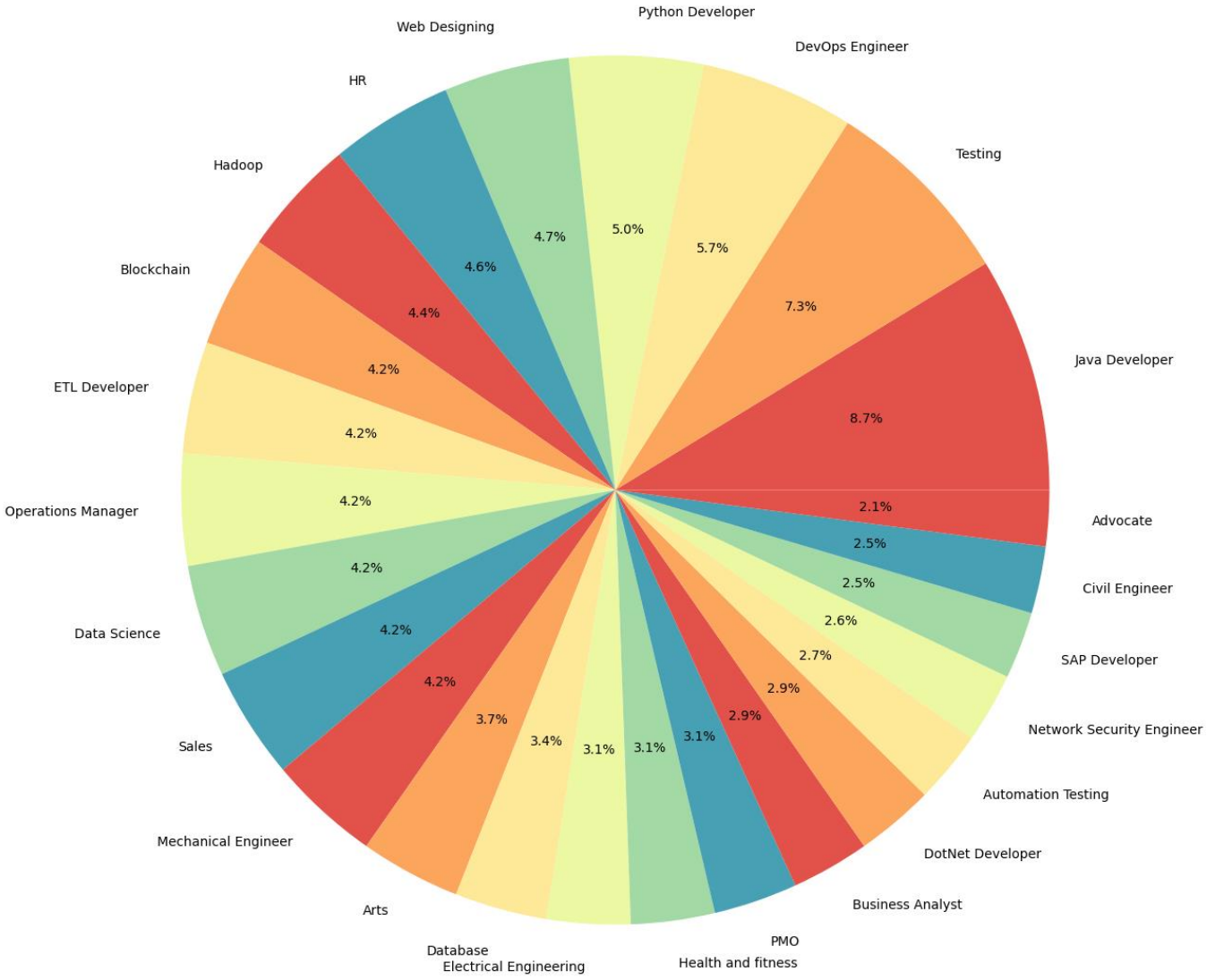
```
Category    0
Resume      0
dtype: int64
```

ภาพที่ 2 ตรวจสอบขนาดข้อมูล ชนิดข้อมูล และค่าว่าง

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

```
25
```

ภาพที่ 3 แสดงจำนวนประเภทงานที่ไม่ซ้ำกัน



ภาพที่ 4 แผนภูมิวงกลมแสดงจำนวนประวัติย่อของแต่ละประเภทงาน

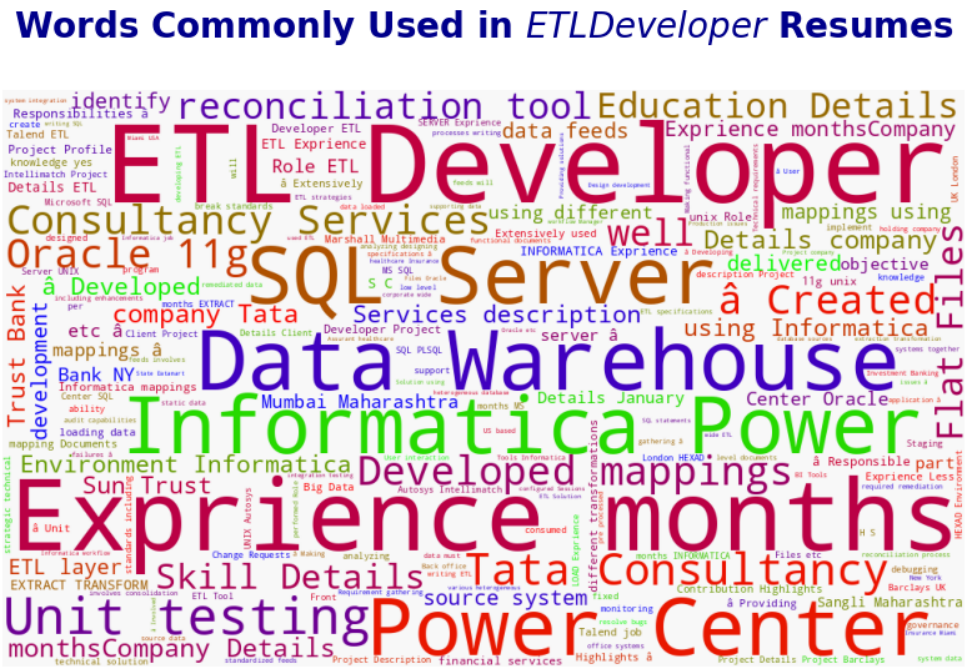
พบว่า 3 อันดับประเภทงานที่มีประวัติย่อมากที่สุด ได้แก่

1. Java Developer (84 ประวัติย่อ)

2. Testing (70 ประวัติย่อ)

3. DevOps Engineer (55 ประวัติย่อ)

โดยแสดงคำที่ใช้บ่อยที่สุดของประวัติย่อในแต่ละประเภทงาน โดยยกตัวอย่างมา 3 ประเภทงาน ดังภาพที่ 5, 6, 7



ภาพที่ 5 คำที่ใช้บ่อยที่สุดใน resume ของประเภทงาน ETL Developer

ภาพที่ 6 คำที่ใช้บ่อยที่สุดใน resume ของประเภทงาน Advocate

ภาพที่ 7 คำที่ใช้บ่อยที่สุดใน resume ของประเภทงาน Data Science

แสดงสถิติเชิงบรรยายทางคณิตศาสตร์ เพื่อดูจำนวนตัวอักษรในประวัติย่อ และการกระจายตัว โดยมีรายละเอียดดังนี้

- 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% ของประวัติย่อมีความยาวน้อยกว่าหรือเท่ากับ 1,217 คำ
- 50% หมายถึง 50% ของประวัติย่อมีความยาวน้อยกว่าหรือเท่ากับ 2,355 คำ
- 75% หมายถึง 75% ของประวัติย่อมีความยาวน้อยกว่าหรือเท่ากับ 4,073 คำ
- max หมายถึง ความยาวประวัติย่อที่ยาวที่สุดคือ 14,816 คำ

ขั้นตอนที่ 3 : การเตรียมข้อมูล (Data Preparation)

ในขั้นตอนการเตรียมข้อมูล ผู้วิจัยได้ใช้ชุดข้อมูลของประวัติย่อ และประเภทงานที่ผ่านการทำ labelling เรียบร้อยแล้ว จากนั้นนำชุดข้อมูลนี้มาเข้าสู่กระบวนการประมวลผลภาษาธรรมชาติ (NLP) โดยมีขั้นตอนดังต่อไปนี้

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

ตาราง 2 แสดงข้อมูลหลังลบคำที่ไม่สำคัญ และตัวอักษรพิเศษออก

```
1 df1['cleaned_resume'] = df1['Resume'].apply(lambda x: clean_function(x))
2 df1.head()
```

	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...

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

ตาราง 3 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	Dot Net Developer
10	ETL Developer
11	Electrical Engineering
12	HR
13	Hadoop
14	Health and fitness

ตางราง 3 (ต่อ)

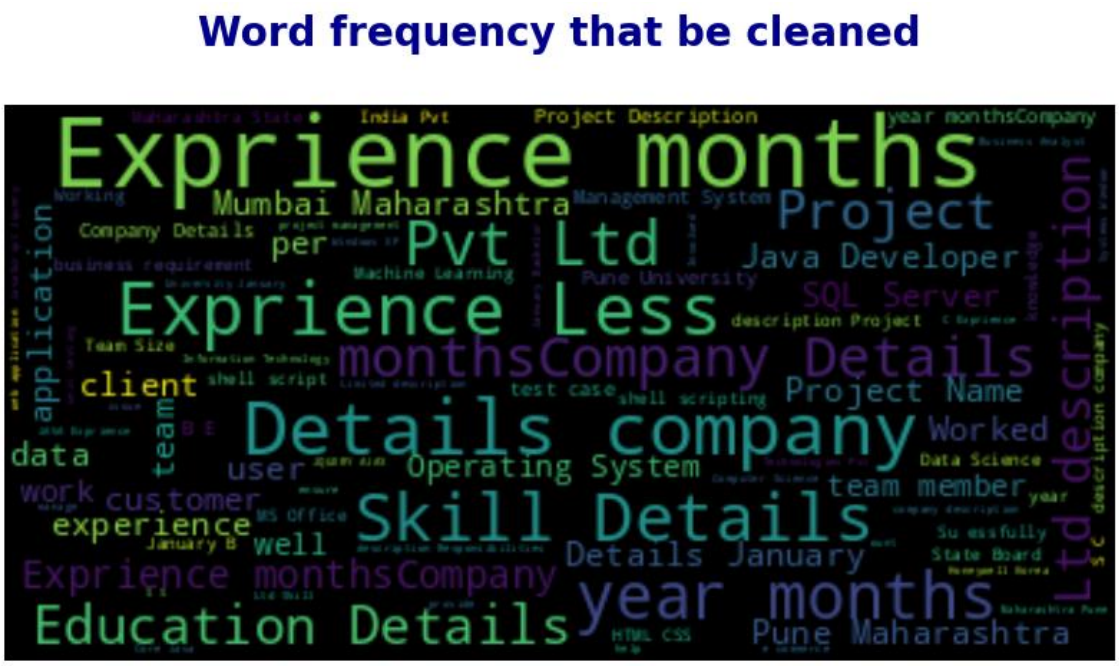
Class	Category Name
15	Java Developer
16	Mechanical Engineer
17	Network Security Engineer
18	Operations Manager
19	PMO
20	Python Developer
21	SAP Developer
22	Sales
23	Testing
24	Web Designing

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

4. ใช้ฟังก์ชัน FreqDist เพื่อดูความถี่ของคำทั้งหมดในข้อความ พบว่าคำว่า Experience มีมากที่สุด จำนวน 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)]
```

ภาพที่ 8 คำที่พบมากที่สุดในประวัติย่อ 30 คำแรก



ภาพที่ 9 Word Cloud ของคำที่พบมากที่สุดในประวัติย่อหลังลบคำที่ไม่สำคัญออก

ขั้นตอนที่ 4 : การสร้างแบบจำลอง (Modeling)

ตรวจสอบข้อมูลให้แน่ใจว่าข้อมูลสะอาด พร้อมนำไปเข้ากระบวนการฝึกอบรม จากนั้นแปลงข้อความเป็นเวกเตอร์ตัวเลขด้วยฟังก์ชัน TF-IDF Vectorization

```
1 df2['Resume'][100]
```

```
'Skills: Natural Languages: Proficient in English, Hindi and Marathi. Computer skills: Proficient with MS-Office, Internet
\r\nJanuary 2015 to January 2018 LLB Law Mumbai, Maharashtra Mumbai university\r\nJanuary 2015 B.M.M  Mumbai, Maharashtra
University\r\n H.S.C   Asmita Girls junior College, Maharashtra Board\r\n S.S.C Vidya Bhawan  Maharashtra Board\r\nAdvocat
urnalist\r\nSkill Details \r\nCompany Details \r\ncompany - Criminal lawyer (law firm)\r\ndescription - '
```

```
1 df2['cleaned_resume'][100]
```

```
'Skills Natural Languages Proficient in English Hindi and Marathi Computer skills Proficient with MS Office Internet opera
ry 2015 to January 2018 LLB Law Mumbai Maharashtra Mumbai university January 2015 B M M Mumbai Maharashtra S K Somaiya Col
Asmita Girls junior College Maharashtra Board S S C Vidya Bhawan Maharashtra Board Advocate Llb student and Journalist Ski
ompany Criminal lawyer law firm description '
```

ภาพที่ 10 แสดงข้อความก่อนและหลังทำความสะอาด

TF-IDF หรือ Term Frequency – Inverse Document Frequency [3] 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) ถ้าหากคำคำไหนถูกพูดถึงอยู่บ่อย ๆ ในเอกสารนั้น ๆ จะมีความเป็นไปได้สูงว่าคำนั้นมีความเกี่ยวข้องกับใจความสำคัญของเอกสารนั้น ๆ มาก ดังสมการ 1

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

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

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

เมื่อนำการคำนวณทั้งสองส่วนมารวมกัน เราจะได้การคำนวณ TF-IDF ดังต่อไปนี้

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

ผู้วิจัยแบ่งชุดข้อมูลออกเป็นชุดข้อมูลสำหรับการฝึกแบบจำลอง (training dataset) 80% และชุดข้อมูลสำหรับการทดสอบแบบจำลอง (testing dataset) 20% random state เท่ากับ 42 และสร้างแบบจำลองเพื่อประเมินประสิทธิภาพความถูกต้องและความแม่นยำ 8 แบบ ได้แก่

1. Support Vector Classification (SVC) ระยะห่างของเวกเตอร์จากไฮเปอร์เพลนเรียกว่าระยะขอบ (margin) ซึ่งเป็นการแยกเส้นไปยังจุดคลาสที่ใกล้ที่สุด เราต้องการเลือกไฮเปอร์เพลนที่เพิ่มระยะขอบระหว่างคลาสให้สูงสุด กราฟด้านล่างแสดงระยะขอบที่ดีและระยะขอบที่ไม่ดี ซึ่ง Margin นี้แบ่งออกเป็น 2 ประเภทคือ Soft Margin และ Hard Margin [5], [6], [7]

2. Logistic Regression เป็นโมเดลการเรียนรู้ของเครื่อง (Machine Learning) ประเภทการจำแนกประเภท (Classification) ที่ใช้สำหรับจำแนกข้อมูลออกเป็นสองคลาส โดยใช้สมการเชิงเส้นในการประมาณความน่าจะเป็นที่ข้อมูลจะอยู่ในคลาสใดคลาสหนึ่ง จะอยู่ระหว่าง 0 ถึง 1 ข้อดีคือ เข้าใจง่าย คำนวณได้รวดเร็ว สามารถประยุกต์ใช้กับปัญหาการจำแนกประเภทได้หลากหลาย

3. Random Forest คือ สร้างแบบจำลองจาก 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 [8]

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

5. Gradient Boosting เป็นโมเดลการเรียนรู้ของเครื่องประเภท Ensemble Learning ที่รวมโมเดลย่อย (Weak Learners) จำนวนมากเข้าด้วยกันเพื่อสร้างโมเดลที่มีประสิทธิภาพมากขึ้น Gradient Boosting ทำงานโดยสร้างโมเดลย่อยขึ้นมาทีละโมเดล โดยโมเดลย่อยแต่ละโมเดลจะพยายามปรับปรุงความผิดพลาดของโมเดลก่อนหน้า ซึ่งโมเดลย่อยแต่ละโมเดลจะสร้างขึ้นมาโดยใช้ Loss Function เพื่อวัดความผิดพลาดของโมเดลก่อนหน้า จากนั้นจะปรับน้ำหนักของโมเดลย่อยนั้นเพื่อให้ความผิดพลาดลดลง

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

7. Gaussian Naïve Bayes เป็นแบบจำลองการเรียนรู้ของเครื่อง (Machine Learning) ที่ใช้ในการจำแนกประเภทของข้อมูลโดยใช้หลักการของ Bayes Theorem และประเมินความน่าจะเป็น (Probability) ของคลาสที่แตกต่างกันตามคุณลักษณะ (Features) ของข้อมูลนำเข้า (Input data) แบบจำลอง Gaussian Naïve Bayes นั้นสามารถใช้ในการจำแนกประเภทของข้อมูลที่มีคุณลักษณะที่เป็นตัวแปรสุ่ม (Continuous variables) โดยที่มีการสมมติว่าค่าความน่าจะเป็นของคุณลักษณะในแต่ละคลาสมีการกระจายแบบ Gaussian (Normal Distribution) ซึ่งเป็นที่มาของคำว่า "Gaussian" ในชื่อของแบบจำลองนี้

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

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

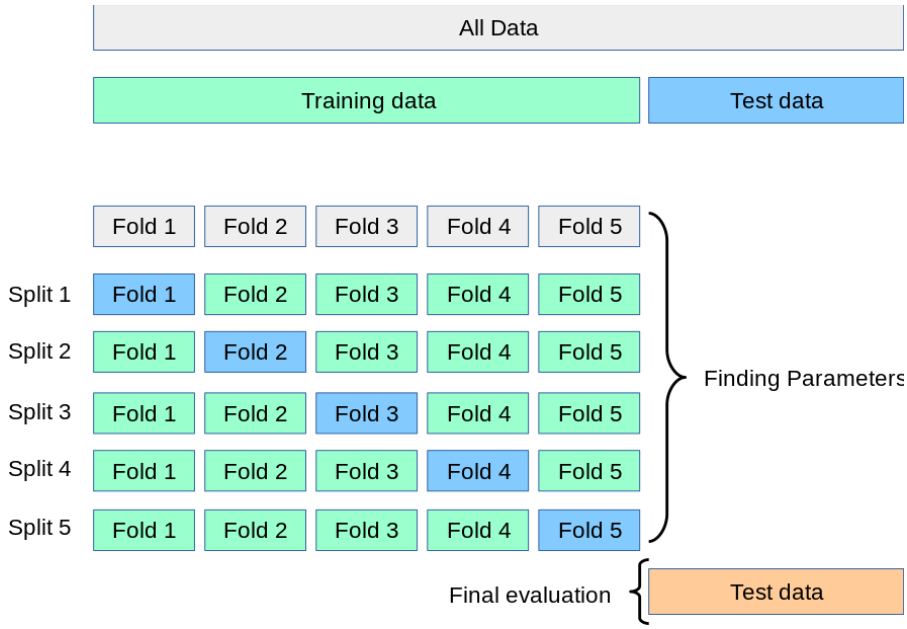
ขั้นตอนที่ 5 : การทดสอบประสิทธิภาพของแบบจำลอง (Cross Validation Model)

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 จากนั้นจึงคำนวณค่าความคลาดเคลื่อนของแบบจำลองจากข้อมูลใน fold ที่ 1

- ในการทดลองครั้งที่สอง ช้ fold ที่ 2 สำหรับทดสอบแบบจำลอง ในขณะที่ใช้ fold ที่ 1, 3, 4, และ 5 สำหรับฝึกอบรมแบบจำลอง จากนั้นจึงคำนวณค่าความคลาดเคลื่อนของแบบจำลองจากข้อมูลใน fold ที่ 2

ทำซ้ำกระบวนการนี้ทั้งหมด 5 ครั้ง โดยสุ่มเลือกชุดย่อยสำหรับทดสอบในแต่ละครั้ง ค่าความคลาดเคลื่อนของแบบจำลองเฉลี่ยจาก 5 ครั้ง จะเป็นค่าความคลาดเคลื่อนของแบบจำลองที่แท้จริงนั่นเอง



ภาพที่ 11 กระบวนการทำงานของ Cross Validation แบบ K-fold [4]

ขั้นตอนที่ 6 : การประเมินแบบจำลอง (Evaluation)

ในการประเมินผลการทดลองการคัดกรองผู้สมัครจากประวัติย่อ เพื่อเปรียบเทียบประสิทธิภาพของแบบจำลองต่าง ๆ ที่ใช้ในการทดลอง ผู้วิจัยเลือกใช้ ค่า Accuracy, ค่า F1 score เป็นตัวชี้วัดการประเมินผลของแบบจำลอง อีกทั้งใช้ Cross Validation แบบ 10 fold เพื่อประเมินค่าความคลาดเคลื่อนของตัวชี้วัดต่าง ๆ เช่น accuracy, precision, recall, และ F1 score อีกด้วย

- Accuracy อัตราส่วนของการทำนายที่ถูกต้องทั้งหมดต่อการทำนายทั้งหมด โดยคำนวณจาก

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

- Precision อัตราส่วนของการทำนายที่ถูกต้องต่อการทำนายทั้งหมดที่เป็นคลาสนั้น ๆ โดยคำนวณจาก

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

- Recall อัตราส่วนของการทำนายที่ถูกต้องต่อข้อมูลจริงทั้งหมดที่เป็นคลาสนั้น ๆ โดยคำนวณจาก

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

- F1 score ค่าเฉลี่ยแบบ harmonic mean ระหว่าง precision และ recall โดยคำนวณจาก

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

ขั้นตอนที่ 7 : การหาคุณลักษณะที่สำคัญ (Feature Important)

ตัวแปรหรือคุณลักษณะที่สำคัญในชุดข้อมูลของงานวิจัยนี้คือคำต่าง ๆ ที่อยู่ในประวัติย่อ เพื่อทราบว่าคำใดที่ส่งผลต่อการจำแนกประเภทงานซึ่งจะนำไปสู่การคัดกรองผู้สมัครต่อไปนั้น โดยทำการสร้างโมเดล Random Forest โดยใช้ข้อมูลที่ถูกสกัดคุณลักษณะด้วย TF-IDF จากนั้นฝึกในแบบจำลองด้วยชุดข้อมูลที่ใช้ในการเรียนรู้ (Training Data) แล้วคำนวณค่าความสำคัญ (Feature Important) มาแสดงผลเป็นกราฟเพื่อดูว่าคุณลักษณะหรือคำใดที่มีความสำคัญมากที่สุด

ผลการวิจัยและอภิปรายผลการวิจัย

เมื่อนำข้อมูลไปฝึกอบรมแล้ว ได้ผลการทดลองดังนี้

- **ค่าความถูกต้อง (Accuracy score)** ของแต่ละแบบจำลองแยกตาม training set และ test set

```
Accuracy of OneVsRestClassifier(estimator=SVC()) on training set : 1.0
Accuracy of OneVsRestClassifier(estimator=SVC()) on test set : 0.9948186528497409


Accuracy of OneVsRestClassifier(estimator=LogisticRegression()) on training set : 0.9986996098829649
Accuracy of OneVsRestClassifier(estimator=LogisticRegression()) on test set : 0.9792746113989638


Accuracy of OneVsRestClassifier(estimator=RandomForestClassifier()) on training set : 1.0
Accuracy of OneVsRestClassifier(estimator=RandomForestClassifier()) on test set : 0.9844559585492227


Accuracy of OneVsRestClassifier(estimator=KNeighborsClassifier()) on training set : 0.9856957087126138
Accuracy of OneVsRestClassifier(estimator=KNeighborsClassifier()) on test set : 0.9585492227979274


Accuracy of OneVsRestClassifier(estimator=GradientBoostingClassifier()) on training set : 1.0
Accuracy of OneVsRestClassifier(estimator=GradientBoostingClassifier()) on test set : 0.9844559585492227


Accuracy of OneVsRestClassifier(estimator=AdaBoostClassifier()) on training set : 1.0
Accuracy of OneVsRestClassifier(estimator=AdaBoostClassifier()) on test set : 0.9948186528497409


Accuracy of OneVsRestClassifier(estimator=GaussianNB()) on training set : 1.0
Accuracy of OneVsRestClassifier(estimator=GaussianNB()) on test set : 0.9844559585492227


Accuracy of OneVsRestClassifier(estimator=DecisionTreeClassifier()) on training set : 1.0
Accuracy of OneVsRestClassifier(estimator=DecisionTreeClassifier()) on test set : 0.9844559585492227
```

ภาพที่ 12 แสดงค่า accuracy ของแต่ละแบบจำลอง แยกตาม training set และ test set

- **ประสิทธิภาพของการจำแนกประเภท** โดยพิจารณาจากค่าความถูกต้อง (Accuracy), ค่าความแม่นยำ (Precision), ค่าความไว (Recall), ค่า f1-score คือค่าเฉลี่ยแบบระหว่าง precision และ recall ซึ่งค่า F1 score ที่สูง หมายถึง แบบจำลองมีความแม่นยำและความไวที่ดี ซึ่งแยกแต่ละคลาสและแต่ละแบบจำลอง ดังนี้

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

****

ภาพที่ 13 แสดงค่า precision, recall, f1 score ของแต่ละคลาสของแบบจำลอง SVC

```
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

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

ภาพที่ 14 แสดงค่า precision, recall, f1 score ของแต่ละคลาสของแบบจำลอง Logistic Regression

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

****

ภาพที่ 15 แสดงค่า precision, recall, f1 score ของแต่ละคลาสของแบบจำลอง Random Forest

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

****

ภาพที่ 16 แสดงค่า precision, recall, f1 score ของแต่ละคลาสของแบบจำลอง K-Nearest Neighbors

```
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

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

ภาพที่ 17 แสดงค่า precision, recall, f1 score ของแต่ละคลาสของแบบจำลอง Gradient Boosting

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

****

ภาพที่ 18 แสดงค่า precision, recall, f1 score ของแต่ละคลาสของแบบจำลอง AdaBoost Classifier

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

****

ภาพที่ 19 แสดงค่า precision, recall, f1 score ของแต่ละคลาสของแบบจำลอง Gaussian Naïve Bayes

```
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

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

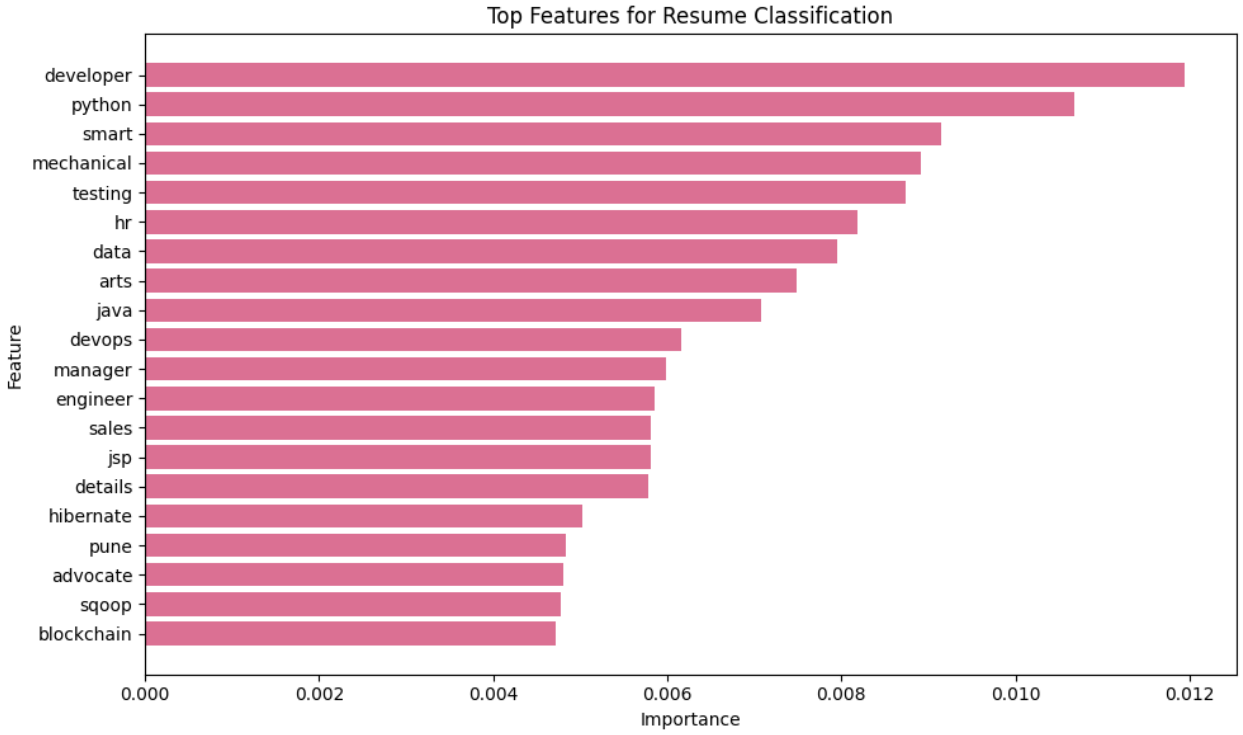
ภาพที่ 20 แสดงค่า precision, recall, f1 score ของแต่ละคลาสของแบบจำลอง Decision Tree

- **ค่า Cross Validation** ของแต่ละแบบจำลองเรียงตามลำดับมากที่สุดไปน้อยที่สุด ผลลัพธ์ดังนี้

ตาราง 4 แสดงค่า Cross Validation ของแบบจำลอง

Model	Cross Validation Result
SVC	0.995
Gradient Boosting	0.992
Random Forest	0.991
AdaBoost Classifier	0.991
Logistic Regression	0.99
Decision Tree	0.987
Gaussian Naïve Bayes	0.984
K-Nearest Neighbors	0.971

- **ผลของคุณลักษณะที่สำคัญต่อการจำแนกประเภท (Feature Importance)** แสดงตัวแปรที่สำคัญที่ส่งผลต่อการจำแนกประเภทงาน จัดทำด้วยวิธีการสร้างแบบจำลอง Random Forest โดยใช้ข้อมูลที่ผ่านการทำ TF-IDF เพื่อสกัดคำที่สำคัญออกมา จากนั้นนำเข้าสู่กระบวนการฝึกฝนในแบบจำลองด้วยชุดข้อมูลที่ใช้ในการเรียนรู้ (Training Data) แล้วคำนวณค่าความสำคัญ (Feature Important) มาแสดงผลเป็นกราฟเพื่อดูว่าคุณลักษณะหรือคำใดที่มีความสำคัญมากที่สุด 20 อันดับ พบว่าคำว่า developer มีค่า feature important มากที่สุดเท่ากับ 0.0119 ดังภาพที่ 21



ภาพที่ 21 แสดงคำที่มีคุณลักษณะสำคัญต่อการจำแนกประเภทมากที่สุด 20 อันดับแรก (Top 20 Feature Important)

สรุปผลการวิจัย

งานวิจัยนี้นำเสนอวิธีการคัดกรองผู้สมัครจากประวัติย่อด้วยหลักการการเรียนรู้ของเครื่อง (Machine Learning) และการประมวลผลภาษาธรรมชาติ (Natural Language Processing) โดยคัดกรองผู้สมัครจากทักษะ การศึกษา ประสบการณ์การทำงานของแต่ละสายงานได้ในเวลาไม่กี่นาที จึงช่วยลดเวลาในการทำงาน และช่วยลดความผิดพลาดที่เกิดจากมนุษย์ได้

ชุดข้อมูลต้นฉบับในงานวิจัยนี้มาจากเว็บไซต์ kaggle.com เป็น Public Dataset ชื่อUpdated Resume Dataset ประกอบด้วย ประเภทงาน (Category) และ ประวัติย่อ (Resume) ข้อมูลมีขนาด 962 แถว 2 คอลัมน์ มีจำนวนประเภทงานที่ไม่ซ้ำกันทั้งหมด 25 ประเภทที่ผ่านการทำ labelling เรียบร้อยแล้ว

ในขั้นตอนการเตรียมข้อมูล ผู้วิจัยได้นำชุดข้อมูลนี้มาเข้าสู่กระบวนการประมวลผลภาษาธรรมชาติ (NLP) เพื่อลบคำต่าง ๆ ที่ไม่สำคัญในประวัติย่อออก จากนั้นนำไปแปลงข้อมูลประเภทงานจากข้อความให้เป็นข้อมูลตัวเลข โดยใช้คำสั่ง Label Encoding เพื่อให้แต่ละประเภทงานกลายเป็นคลาส และนำเข้าสู่กระบวนการแบ่งข้อความออกเป็นหน่วยเล็ก ๆ (Tokenization) เพื่อช่วยในการทำความเข้าใจบริบทของข้อความ จากนั้น ตัด stop words หรือ คำที่เจอบ่อย ๆ แต่ไม่สื่อความหมายออก สุดท้าย ทำ Lemmatization หรือการเปลี่ยนรูปคำให้อยู่ในรูปแบบของคำดั้งเดิมหรือคำกริยาช่องที่ 1 เพื่อให้อยู่ในรากศัพท์เดียวกัน

แบ่งชุดข้อมูลออกเป็นชุดข้อมูลสำหรับการฝึกแบบจำลอง (training dataset) 80% และชุดข้อมูลสำหรับการทดสอบแบบจำลอง (testing dataset) 20% random state เท่ากับ 42 และสร้างแบบจำลองเพื่อประเมินประสิทธิภาพความถูกต้องและความแม่นยำ 8 แบบ ได้แก่ Support Vector Classification (SVC), Logistic Regression, Random Forest, K-Nearest Neighbors (KNN), Gradient Boosting, AdaBoost Classifier, Gaussian Naïve Bayes, Decision Tree

จากผลการทดลองพบว่าค่าความถูกต้อง (Accuracy score) ของแบบจำลอง Support Vector Classification (SVC) และแบบจำลอง AdaBoost Classifier มีค่ามากที่สุด เท่ากับ 0.994 เมื่อเทียบกับแบบจำลองอื่น ๆ แต่แบบจำลอง Support Vector Classification (SVC) มีค่า Cross Validation มากที่สุดคือ 0.995 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 เมื่อดูว่าคำไหนส่งผลต่อการจำแนกประเภทงานมากที่สุด พบว่าคำว่า developer มีค่า feature important มากที่สุดเท่ากับ 0.0119

ผู้วิจัยตั้งสมมติฐานเรื่องประเภทแบบจำลองก่อนทำการทดลองว่าแบบจำลอง Random Forest และแบบจำลองประเภท Boosting น่าจะเป็นแบบจำลองที่มีค่าความถูกต้อง และค่าความแม่นยำดีที่สุด แต่หลังจากทำการทดลองแล้วนั้นพบว่าไม่ได้เป็นอย่างที่คิดไว้ กลับกลายเป็นแบบจำลอง SVC ดีที่สุด อาจเป็นเพราะชุดข้อมูลที่ใช้ไม่ได้มีความซับซ้อนมากจนต้องใช้แบบจำลองประเภท Random Forest หรือ Boosting ถ้างานที่เรากำลังเป็นข้อมูลที่ไม่ซับซ้อนการมใช้แบบจำลองแบบ SVC ก็สามารถเกิดความแม่นยำของการทำนายได้เพียงพอแล้ว

เพื่อเพิ่มประสิทธิภาพของการคัดกรองผู้สมัครจากประวัติย่อด้วยหลักการเรียนรู้ของเครื่อง ผู้วิจัยขอเสนอแนะเพิ่มเติมว่าสามารถนำไปต่อยอดได้ กล่าวคือสามารถนำระบบการคัดกรองผู้สมัครจากประวัติย่อด้วยการเรียนรู้ของเครื่องเชื่อต่อกับหน้าเว็บไซต์หรือแอปพลิเคชันบนมือถือเพื่อให้ผู้ใช้งานหรือฝ่ายคัดสรรบุคลากรอัปโหลดประวัติย่อของผู้สมัครแบบเรียลไทม์เพื่อนำไปเข้าสู่กระบวนการของการคัดสรรได้อย่างมีประสิทธิภาพและรวดเร็วแม่นยำมากขึ้น

กิตติกรรมประกาศ

การจัดทำวิจัยได้รับการสนับสนุนจากบัณฑิตวิทยาลัย มหาวิทยาลัยศรีนครินทรวิโรฒ ในการนำเสนอผลงานวิจัย ผู้วิจัยจึงขอขอบคุณมา ณ ที่นี้

เอกสารอ้างอิง

[1] HREX.asia. (2019, March 13). การสรรหาบุคลากร (Recruitment) เพื่อเข้าร่วมงานกับองค์กร. [Online]. Available: https://th.hrnote.asia/recruit/190313-recruitment/

[2] Gaurav Dutta. Resume Dataset. [Online]. Available: https://www.kaggle.com/datasets/gauravduttakiit/resume-dataset/data

[3] Patipan Prasertsom. (2020, October 1). สกัดใจความสำคัญของข้อความด้วยเทคนิคการประมวลผลทางภาษาเบื้องต้น: TF-IDF, Part 1. [Online]. Available: https://bdi.or.th/big-data-101/tf-idf-1/

[4] scikit-learn. 3.1. Cross-validation: evaluating estimator performance. [Online]. Available: https://scikit-learn.org/stable/modules/cross_validation.html

[5] ชิตพงษ์ กิตตินราดร. (2020, January). Support Vector Machines. [Online]. Available: https://guopai.github.io/ml-blog08.html

[6] Premanand S . (2021, June 16). The A-Z guide to Support Vector Machine. [Online]. Available: https://www.analyticsvidhya.com/blog/2021/06/support-vector-machine-better-understanding/

[7] Mr. P L. (2018, November 15). SVM อดีตเคยหวานปัจจุบันแอบเซง : Machine Learning 101. [Online]. Available: https://medium.com/mmp-li/svm-อดีตเคยหวานปัจจุบันแอบเซง-machine-learning-101-6008753c780c

[8] Witchapong Daroontham. (2018, November 21). เจาะลึก Random Forest !!!— Part 2 of “รู้จัก Decision Tree, Random Forest, และ XGBoost!!!”. [Online]. Available: https://medium.com/@witchapongdaroontham/เจาะลึก-random-forest-part-2-of-รู้จัก-decision-tree-random-forest-และ-xgboost-79b9f41a1c1c