

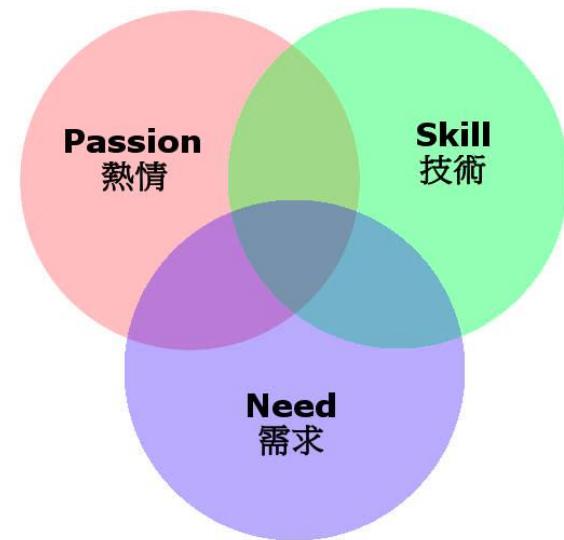


# Data Analytics Research Team (DART) – 2025

陳弘軒 Hung-Hsuan Chen  
Computer Science & Information  
Engineering  
National Central University

# Mission: data science for the society

- Discover the necessity and problem (**Need**)
- Equip with programing and math skills along with domain knowledge to solve the problem (**skill**)
- Willing to practice and make it happen (**Passion**)
- 研究應有所本，不單為研究而研究
- 及早開始研究對學生的好處：產生學生時代的「代表作」
  - 好的論文有助於申請出國留學、好的專案有助於求職



# Recent research/project direction

- Develop machine learning models that are
  - Faster (shorter training or inference time)
  - More accurate
  - Better (under certain conditions)
- Apply machine learning to applications
  - Smart sport (精準運動科學)
  - Privacy-preserving machine learning
  - Search engines & recommender systems
  - PM2.5 prediction & sensor malfunction prediction
  - Traffic prediction
  - Personality traits and personality prediction
  - Clip search within videos
  - Log analysis

# Table of contents

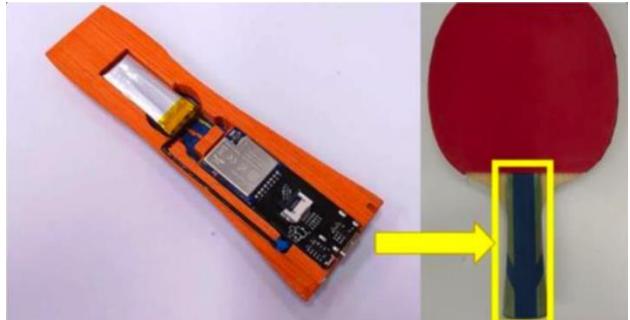
- Recent graduate projects
- Recent undergraduate projects (大學專題)

# Recent graduate projects

# 精準運動科學 – 桌球 (2023 – 2027)

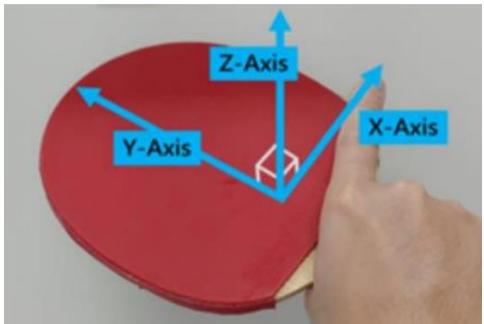
- 結合AI的揮拍者能力分析系統
  - 資料蒐集
  - 挥拍者個人運動能力分析
- 戰術分析
  - 資料蒐集與標註
  - 選手打法分析

# 精準運動科學 - 桌球 (結合AI的揮拍者能力分析系統 - 資料蒐集)



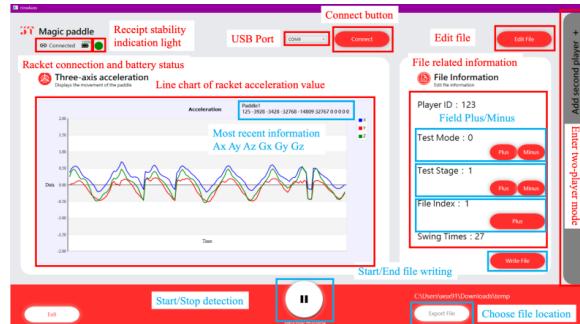
智慧球拍內嵌感測器示意圖

- 單點回擊



x,y,z軸向定義

- 兩側移動回擊



程式紀錄受測者之揮拍資訊及  
揮拍模式



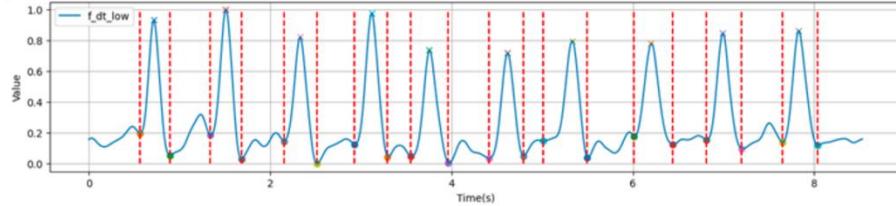
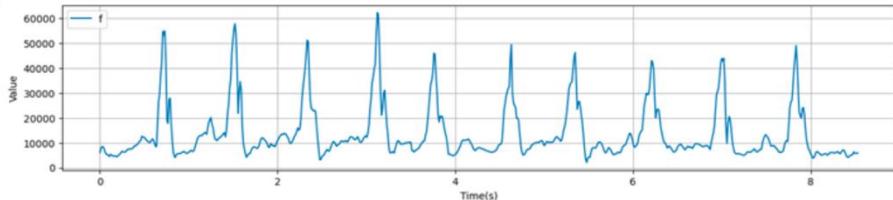
C.-Y. Chou et al., Springer Nature Scientific Data 2025

P.-C. Cheng et al., AIoT 2025

# 精準運動科學 - 桌球 (結合AI的揮拍者能力分析系統)

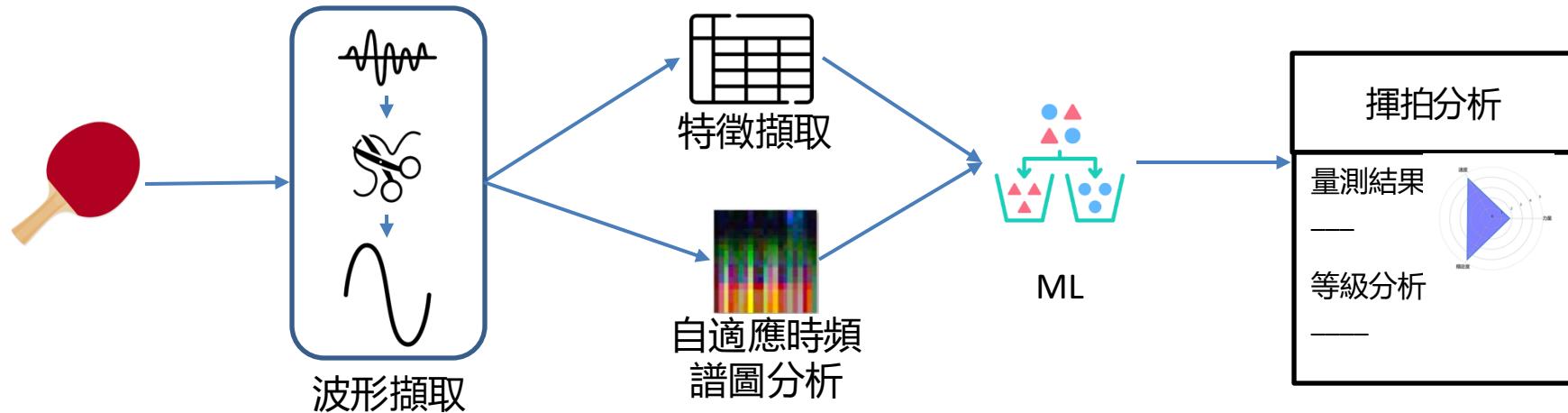


## 口揮拍波形平滑化技術及自動波形切割



## 口揮拍波形AI及大數據分析，產出科學化個人報告：

- (1) 量化揮拍者的揮拍速度、加速度、穩定度數值
- (2) 比較揮拍者的揮拍速度、加速度、穩定度數值在所有受測球員中之百分比
- (3) 挥拍者之等級分析



# 精準運動科學 - 桌球 (戰術分析標註)

Chuang Chih-Yuan vs Ruwen Filus | MS | Singapore Smash 2022 ...

稍後觀看 分享

#WTT SINGAPORE SMASH

PORE SMASH

CHUANG 0 8

FILUS 0 8

更多影片

1:03 / 9:09

YouTube

計分板

2:2

Filus Ruwen 9:10 莊智淵

暫停 暫停

發球

編輯

Filus Ruwen

莊智淵

左 中 右

確認輸出 贏(無錄影) 發球失誤 輸(無錄影) 刪除最後一板

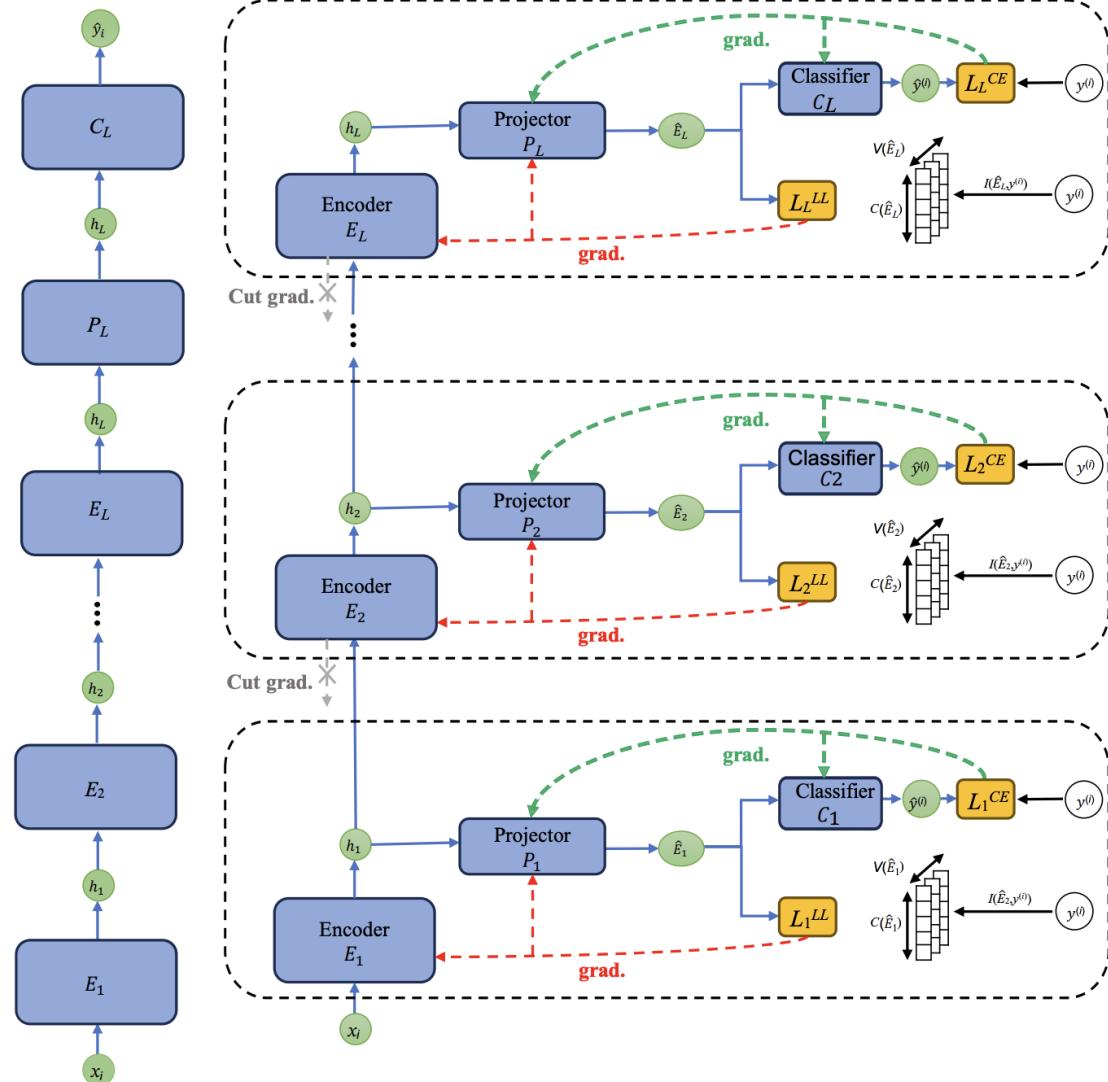
比賽時序情境									
局數	比數	球權	板數	正反拍	技術	旋轉	速度	落點	結果
5	10:9	莊智淵	1	無	無	無	無	贏(無錄影)	莊智淵獲勝
5	9:9	Filus Ruwen	1	無	無	無	無	輸(無錄影)	莊智淵獲勝
5	9:8	Filus Ruwen	1	無	無	無	無	輸(無錄影)	莊智淵獲勝
5	7:9	莊智淵	1	無	無	無	無	贏(無錄影)	莊智淵獲勝
5	6:9	莊智淵	1	無	無	無	無	贏(無錄影)	莊智淵獲勝
5	9:5	Filus Ruwen	1	無	無	無	無	輸(無錄影)	莊智淵獲勝
5	9:4	Filus Ruwen	1	無	無	無	無	輸(無錄影)	莊智淵獲勝

## 口標註廣泛而詳盡

- 口已標註 100,000+ 拍
- 口涵蓋15個級別賽事 (奧運、世錦賽、亞運、世大運、全國賽、大專盃等)
- 口選手涵蓋29個國家
- 口每拍包括：擊球種類 (如：正反手，發球、攻球、擰球)、落點等
- 口原始影片

# DeInfoReg – 實現 backprop 的分段平行化 (2023 – 2025)

- Realizes model parallelism for deep learning; maintains high test accuracies across different networks and open datasets



# DeInfoReg – accurate, efficient, and robust

## Accurate predictions

Batch Size	64	128	256
<b>IMDB</b>			
BP	87.91 $\pm$ 0.44	87.54 $\pm$ 0.29	87.50 $\pm$ 0.42
AL	87.93 $\pm$ 0.37	86.89 $\pm$ 0.73	87.98 $\pm$ 0.48
SCPL	88.16 $\pm$ 0.48	88.89 $\pm$ 0.29	88.69 $\pm$ 0.32
DeInfoReg	<b>89.02 <math>\pm</math> 0.17</b>	<b>89.26 <math>\pm</math> 0.09</b>	<b>88.86 <math>\pm</math> 0.30</b>
<b>AGNews</b>			
BP	91.05 $\pm$ 0.10	90.84 $\pm$ 0.11	90.74 $\pm$ 0.16
AL	88.20 $\pm$ 1.60	86.20 $\pm$ 1.90	89.41 $\pm$ 0.86
SCPL	91.36 $\pm$ 0.21	91.64 $\pm$ 0.07	91.66 $\pm$ 0.21
DeInfoReg	<b>91.91 <math>\pm</math> 0.13</b>	<b>91.95 <math>\pm</math> 0.06</b>	<b>91.90 <math>\pm</math> 0.15</b>
<b>DBpedia</b>			
BP	97.66 $\pm$ 0.05	97.63 $\pm$ 0.01	97.59 $\pm$ 0.02
AL	91.20 $\pm$ 2.37	92.81 $\pm$ 2.02	93.58 $\pm$ 1.62
SCPL	97.35 $\pm$ 0.27	97.47 $\pm$ 0.04	97.58 $\pm$ 0.05
DeInfoReg	<b>97.84 <math>\pm</math> 0.02</b>	<b>97.84 <math>\pm</math> 0.02</b>	<b>97.85 <math>\pm</math> 0.03</b>

## Fast training with multiple GPUs

Batch size	256	512	1024
BP	1x (28.74 sec)	1x (28.44 sec)	1x (28.8 sec)
DeInfoReg (1 GPU)	0.81x	0.85x	0.90x
DeInfoReg (2 GPUs)	1.30x	1.35x	1.35x
DeInfoReg (4 GPUs)	1.41x	1.44x	1.47x

## Robust to noisy data

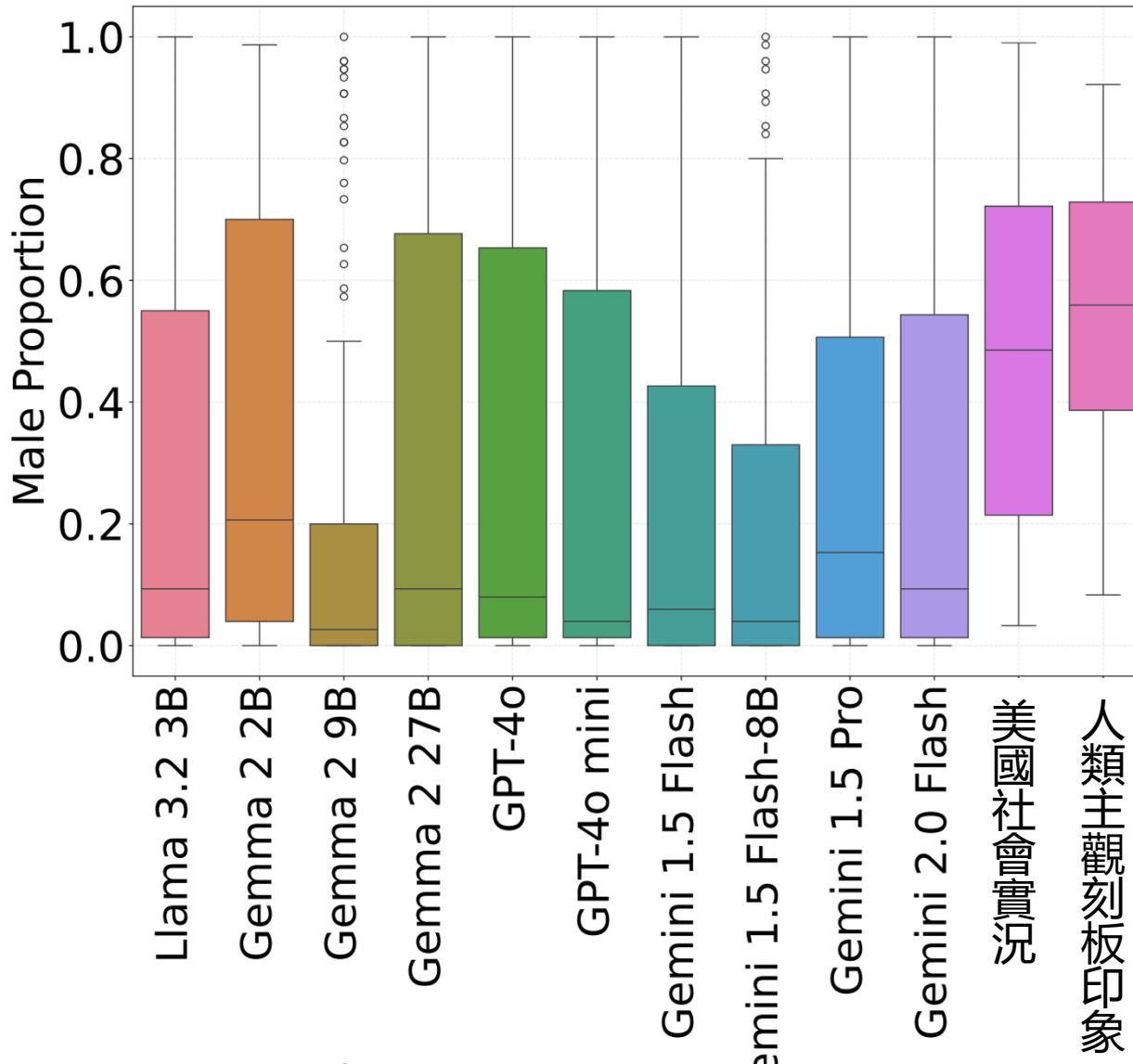
$\theta$ : noisy ratio

$\theta$	0.0	0.2	0.4	0.6	0.8
BP	88.27 $\pm$ 0.48	83.01 $\pm$ 1.2	76.08 $\pm$ 0.13	67.71 $\pm$ 0.4	57.94 $\pm$ 1.75
DeInfoReg	90.15 $\pm$ 0.18	85.46 $\pm$ 0.66	81.28 $\pm$ 1.1	76.36 $\pm$ 5.18	68.1 $\pm$ 8.31

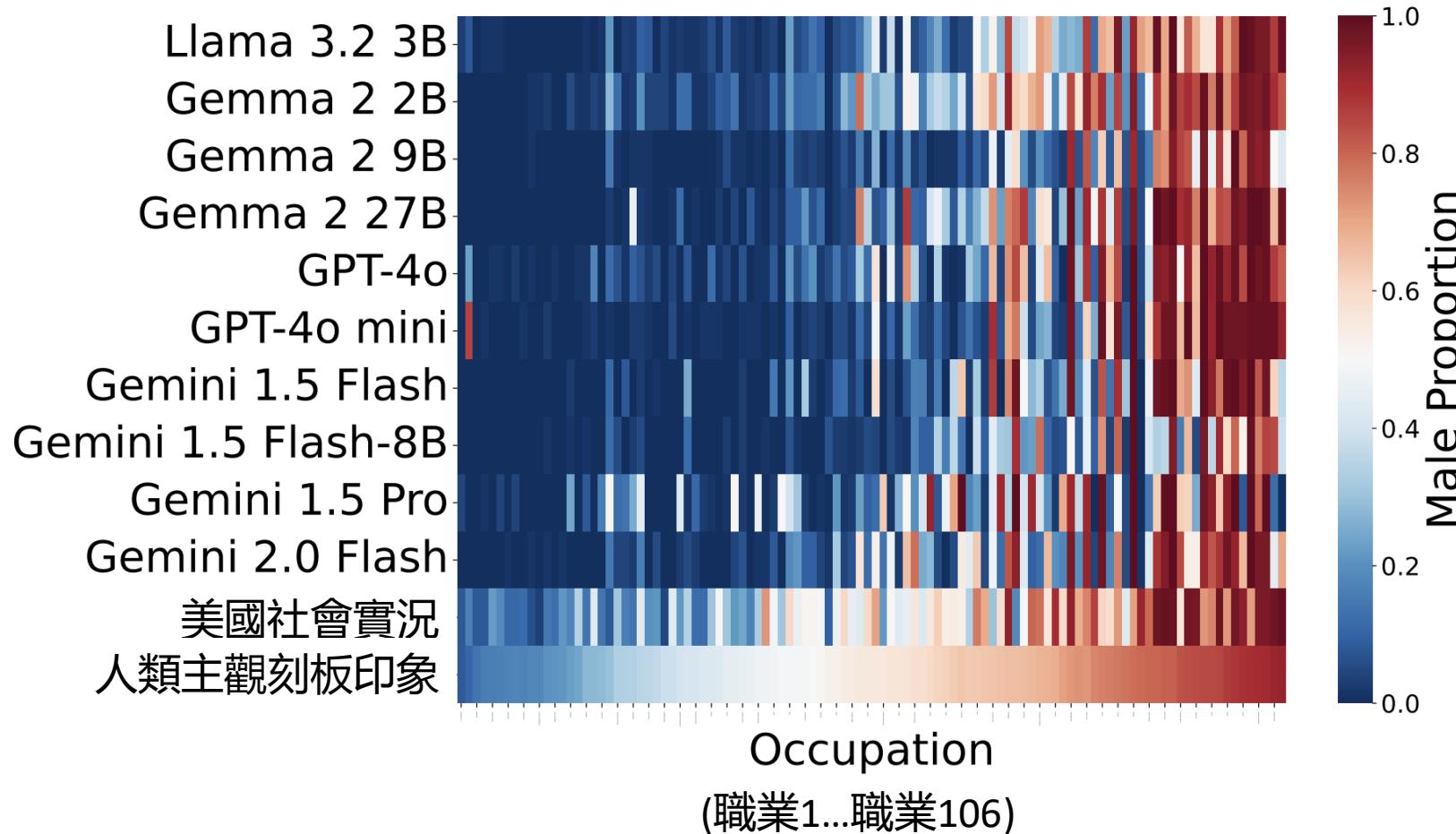
# 量化LLM對「職業-性別」的刻板印象 (2024 – 2025)

- 大規模量化LLM對職業的性別刻板印象
  - 10 LLMs (include proprietary & open source)
  - 106個職業
- 發現1: LLM 嚴重高估各職業的女性比例
- 發現2: 即便如此，人類對於「職業-性別」的刻板印象仍存在於 LLM 中
- 發現3: LLM高估女性的原因可能來自 SFT 與 RLHF

# LLM對「職業-性別」的刻板印象 – 高估各職業的女性比例 (1/2)



# LLM對「職業-性別」的刻板印象 – 高估各職業的女性比例 (2/2)

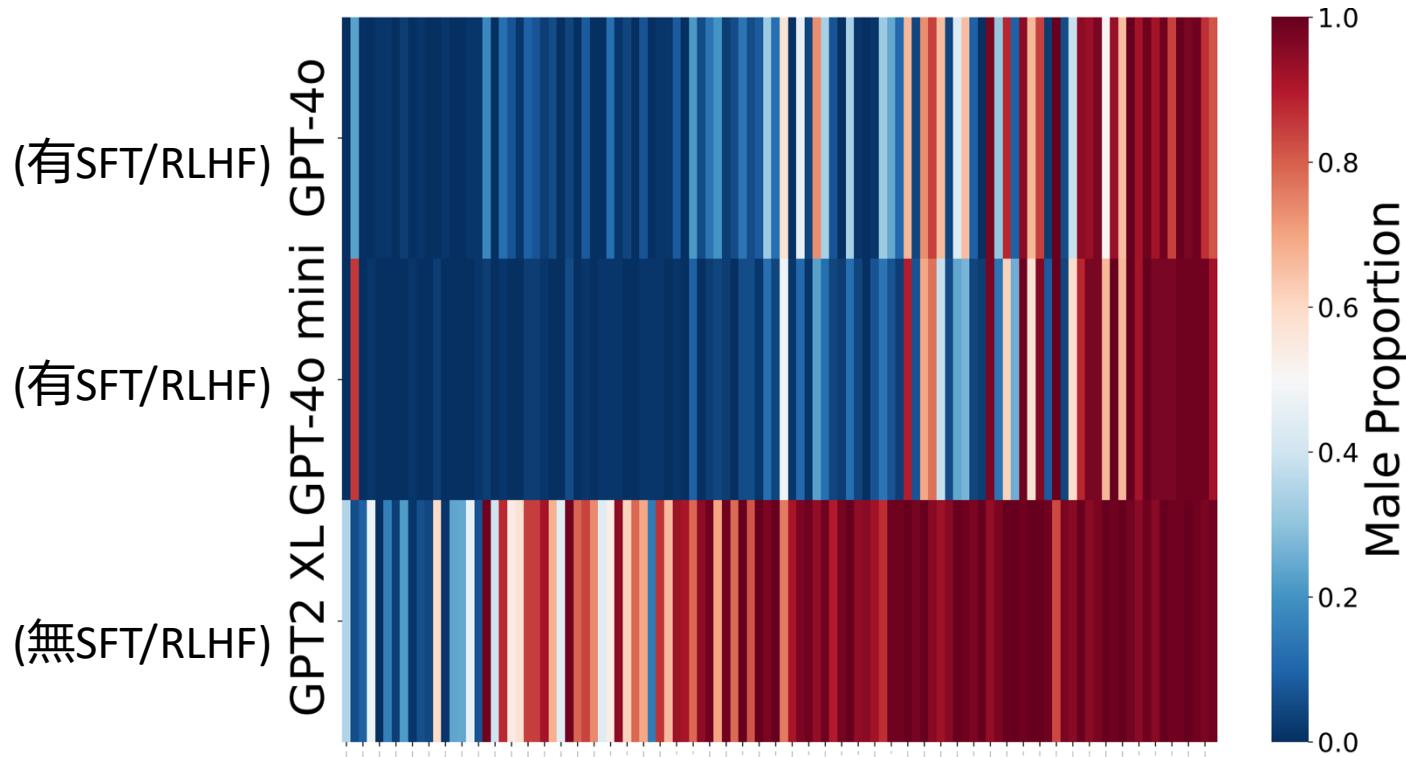


# 以Gender Ratio 排序職業時，LLM與人類刻板印象高度一致

Model	Kendall's Tau	
	美國社會實況	人類主觀刻板印象
Llama3.2 3B	0.6352	<b>0.6975</b>
Gemma2 2B	0.6642	<b>0.7411</b>
Gemma2 9B	0.6125	<b>0.6919</b>
Gemma2 27B	0.6159	<b>0.7201</b>
Gpt-4o mini	0.5939	<b>0.6712</b>
Gpt-4o	0.5375	<b>0.5666</b>
Gemini 1.5 flash	0.5522	<b>0.6364</b>
Gemini 1.5 flash-8B	0.5577	<b>0.6368</b>
Gemini 1.5 pro	0.3368	<b>0.4557</b>
Gemini 2.0 flash	0.5383	<b>0.6362</b>

# LLM高估各職業的女性比例，但職業按 gender ratio 排序卻與人類刻板印象一致？

- 可能成因：Supervised finetuning (SFT) 與 Reinforcement Learning from Human Feedback (RLHF)



# 讓機器學會分類任務的類別編碼 (Contrastive ECOC)

- 分類任務的目標變數一般使用獨熱 (one-hot) 編碼
- 潛在問題：獨熱編碼使得各類別的距離相等，但這不見得是對的設定
  - 例子：若預測目標為「高、中、低」，則「高、中」間的距離應小於「高、低」間的距離
- 我們設計了Contrastive ECOC 方法，讓機器幫類別進行合適的編碼

# 使用 Contrastive ECOC 編碼的效果— 乾淨 vs 經過惡意攻擊的訓練資料

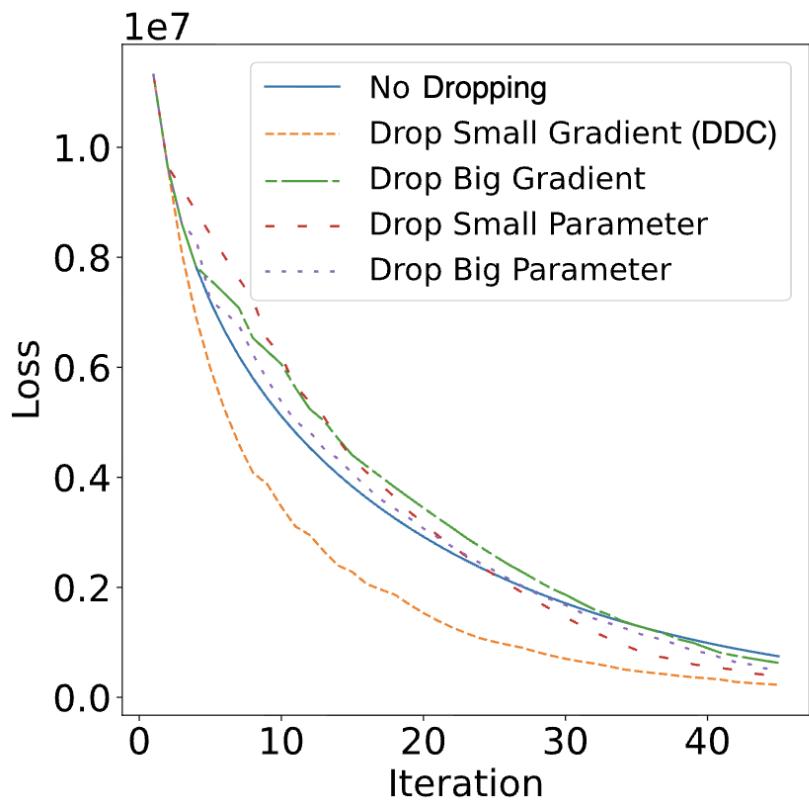
原始資料集	攻擊法	比較的 baselines			三種 Contrastive ECOC方法		
		Standard	SimCLR	ACL-PF	ACL-CFPC	ACL-TFC	
CIFAR-10	CLEAN	86.27%	90.07%	90.1%	<b>90.28%</b>	84.28%	
	FGSM	4.63%	19.96%	39.81%	<b>40.77%</b>	21.94%	
	PGD	0%	0.06%	2.4%	3.21%	<b>14.51%</b>	
Fashion-MNIST	CLEAN	<b>92.7%</b>	92.41%	92.65%	92.64%	92.42%	
	FGSM	29.15%	39.54%	46.83%	53.45%	<b>82.3%</b>	
	PGD	7.75%	3.77%	10.82%	12.65%	<b>35.09%</b>	
GTSRB	CLEAN	<b>92%</b>	90.67%	90.98%	89.67%	57.85%	
	FGSM	36.81%	45.89%	52.26%	<b>52.65%</b>	33.25%	
	PGD	1.79%	7.76%	3.27%	4.67%	<b>12.96%</b>	

- Contrastive ECOC能抵抗 FGSM及PGD攻擊
- 無攻擊時， Contrastive ECOC 的能力與標準方法互有輸贏

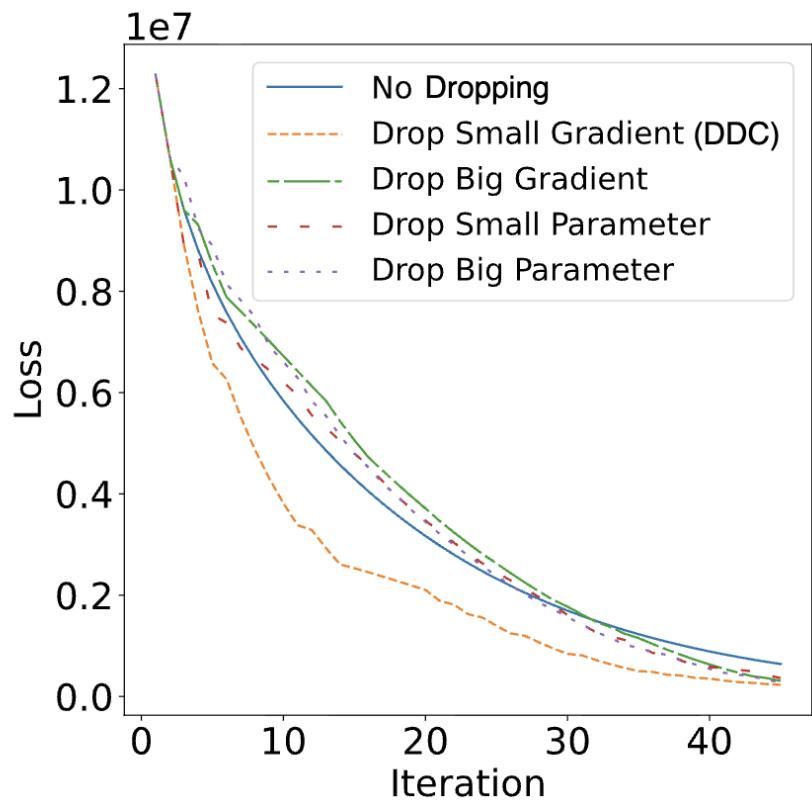
# Dynamic DropConnect: 神經網路訓練時，動態調整每條連線的關閉率

- Dropout/DropConnect: 訓練神經網路時防止過適的標準做法
  - 事先人為定義關閉率 $p$
  - 訓練時每個回合的每條連線皆有 $p$ 機率被關閉
- 問題:  $p$ 是個超參數，需人為設定
- 我們希望由模型自動決定每條連線各自的關閉率
- 實驗設置:
  - 關閉當下低梯度值的邊 / 關閉當下高梯度值的邊 / 關閉當下高權重植的邊 / 關閉當下低權重植的邊

# 優先關閉低梯度的邊較快收斂 (on training data)



(a)  $w_1 = 11.5, w_2 = -10$



(b)  $w_1 = 9, w_2 = -10.5$

# 優先關閉低梯度的邊讓模型更穩健 (on test data)

CIFAR-10

Method	Network Structure	
	AlexNet	VGG
No Dropping	$81.04 \pm 0.04$	$90.46 \pm 0.12$
DDC	<b><math>84.25 \pm 0.09(\uparrow\uparrow)</math></b>	<b><math>90.94 \pm 0.11(\uparrow\uparrow)</math></b>
Dropout	$83.86 \pm 0.09(\uparrow\uparrow)$	$90.66 \pm 0.07(\uparrow\uparrow)$
DropConnect	$83.52 \pm 0.17(\uparrow\uparrow)$	$90.68 \pm 0.08(\uparrow\uparrow)$
Standout	$83.75 \pm 0.04(\uparrow\uparrow)$	$90.64 \pm 0.08(\uparrow)$
Drop Small Parameter	$83.00 \pm 0.10(\uparrow\uparrow)$	$89.88 \pm 0.01(\downarrow\downarrow)$
Drop Big Parameter	$83.81 \pm 0.06(\uparrow)$	$90.45 \pm 0.02(\downarrow)$
Drop Big Gradient	$83.19 \pm 0.16(\uparrow)$	$90.88 \pm 0.03(\uparrow\uparrow)$

CIFAR-100

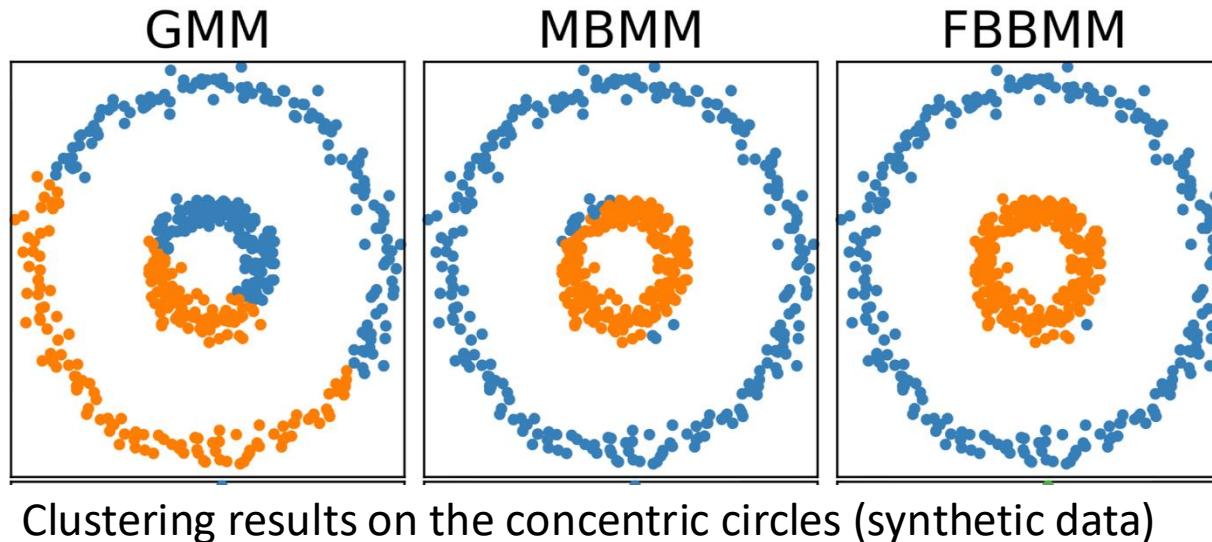
Method	Network Structure	
	AlexNet	VGG
No Dropping	$62.53 \pm 0.50$	$71.38 \pm 0.03$
DDC	$64.06 \pm 0.37(\uparrow\uparrow)$	<b><math>72.09 \pm 0.04(\uparrow\uparrow)</math></b>
Dropout	<b><math>66.53 \pm 0.27(\uparrow\uparrow)</math></b>	$71.69 \pm 0.06(\uparrow\uparrow)$
DropConnect	$63.72 \pm 0.11(\uparrow\uparrow)$	$71.60 \pm 0.21(\uparrow)$
Standout	$63.97 \pm 0.38(\uparrow\uparrow)$	$71.53 \pm 0.31(\uparrow)$
Drop Small Parameter	$63.10 \pm 0.13(\uparrow)$	$71.71 \pm 0.10(\uparrow\uparrow)$
Drop Big Parameter	$63.02 \pm 0.47(\uparrow)$	$71.82 \pm 0.03(\uparrow\uparrow)$
Drop Big Gradient	$62.92 \pm 0.22(\uparrow)$	$71.13 \pm 0.07(\downarrow\downarrow)$

NORB

Method	Network Structure	
	AlexNet	VGG
No Dropping	$92.08 \pm 0.11$	$93.36 \pm 0.03$
DDC	<b><math>92.93 \pm 0.09(\uparrow\uparrow)</math></b>	<b><math>94.20 \pm 0.16(\uparrow\uparrow)</math></b>
Dropout	$92.04 \pm 0.07(\downarrow)$	$93.79 \pm 0.25(\uparrow\uparrow)$
DropConnect	$92.09 \pm 0.14(\uparrow)$	$93.94 \pm 0.19(\uparrow\uparrow)$
Standout	$92.08 \pm 0.07$	$93.81 \pm 0.08(\uparrow\uparrow)$
Drop Small Parameter	$92.42 \pm 0.10(\uparrow\uparrow)$	$94.02 \pm 0.05(\uparrow\uparrow)$
Drop Big Parameter	$92.02 \pm 0.04(\downarrow)$	$93.82 \pm 0.15(\uparrow\uparrow)$
Drop Big Gradient	$91.85 \pm 0.13(\downarrow)$	$94.17 \pm 0.07(\uparrow\uparrow)$

# Multivariate Beta Mixture Model (MBMM) & Flexible Bivariate Beta Mixture Model (FBBMM)

- New probabilistic clustering algorithms
- Gaussian mixture model (GMM): each cluster has to be a Gaussian distribution
- MBMM & FBBMM: allow versatile shapes for each cluster
  - Uni-modal (symmetric or skewed), bi-modal



# Math Information Retrieval (MathIR) (1/2)

- Query: “ $ax^2 + bx + c = 0$ ”
  - Does “ $\alpha\theta^2 - \beta\theta = \gamma$ ” count as a match?
    - If  $x$  is the unknown in Eq1,  $\theta$  is the unknown in Eq2, the two equations are equivalent
    - However, if compute similarity based on text-matching-based methods (e.g., TF-IDF), Eq1 and Eq2 are not similar
- Challenge
  1. MathIR is beyond text-matching
  2. Labeled data (i.e., which pairs of equations are similar) is limited

# Math Information Retrieval (MathIR) (2/2)

- Convert equation into graphs
  - Capture the notation structures
    - Tackles challenge 1 (MathIR is beyond text matching)
- Graph Contrastive Learning (GCL)
  - Generate similar equation pairs based on graph augmentation techniques
    - Tackles challenge 2 (limited labeled data)
- Bpref scores (TangentCFT is a SOTA)

Model	SLT	OPT	F1
TangentCFT	$0.680 \pm 0.0053$	$0.660 \pm 0.0064$	0.670
InfoGraph	$0.691 \pm 0.0066$	$0.685 \pm 0.0070$	0.688
BGRL	<b><math>0.701 \pm 0.0089</math></b>	$0.683 \pm 0.0077$	0.692
GraphCL	$0.685 \pm 0.0090$	<b><math>0.703 \pm 0.0072</math></b>	<b>0.694</b>

# Privacy-preserving machine learning

- 政府開放資料可促進創新，但帶有隱私資訊的資料難以直接開放
- 假名化：將可識別個人的資訊 (e.g., 姓名、身份證字號) 替換為假名或代號
- 假名化 (pseudonymization) 仍會洩露隱私
  - 莊智淵是台灣唯一40+的桌球國手
  - 中央大學資工系學生中來自金門高中僅個位數
- 差分隱私 (Differential privacy):
  - (Layman's language) 資料中加入雜訊，使得難以從資料中推測出具體個人的資訊
  - 但差分隱私會讓資料的實用性降低

# 隱私強化技術：差分隱私與合成資料 (2024)

- 技術檢測項目
- 隱私強化技術演算法適用性分析
- 概念驗證實作

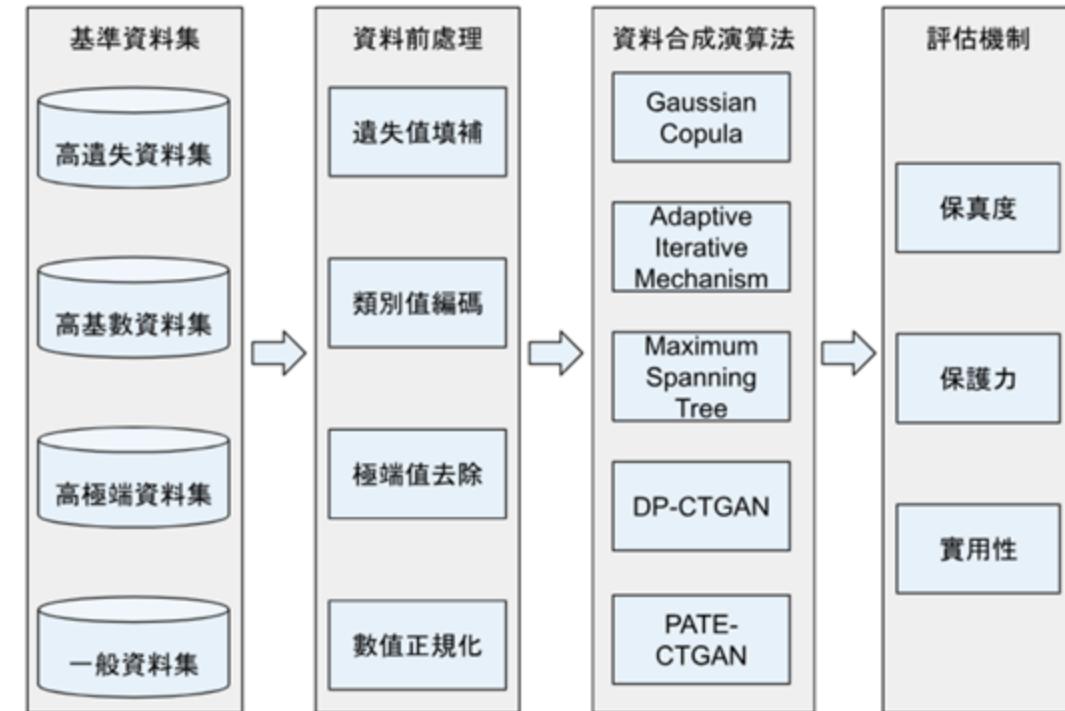
# 隱私強化技術演算法適用性分析 – 基準資料集特性

高遺失：

- 資料內總遺失筆數超過總筆數15%，或
- 內部遺失值超過 15% 的欄位 (column) 超過總欄位數之 1/3

高基數：類別變項基數 (cardinality) 大於 10 的欄位超過總欄位數之 1/3

高極端：偏態係數 (skewness) 絶對值大於 3 的欄位超過總欄位數之 1/3



# 隱私強化技術演算法適用性分析 – 資料前處理

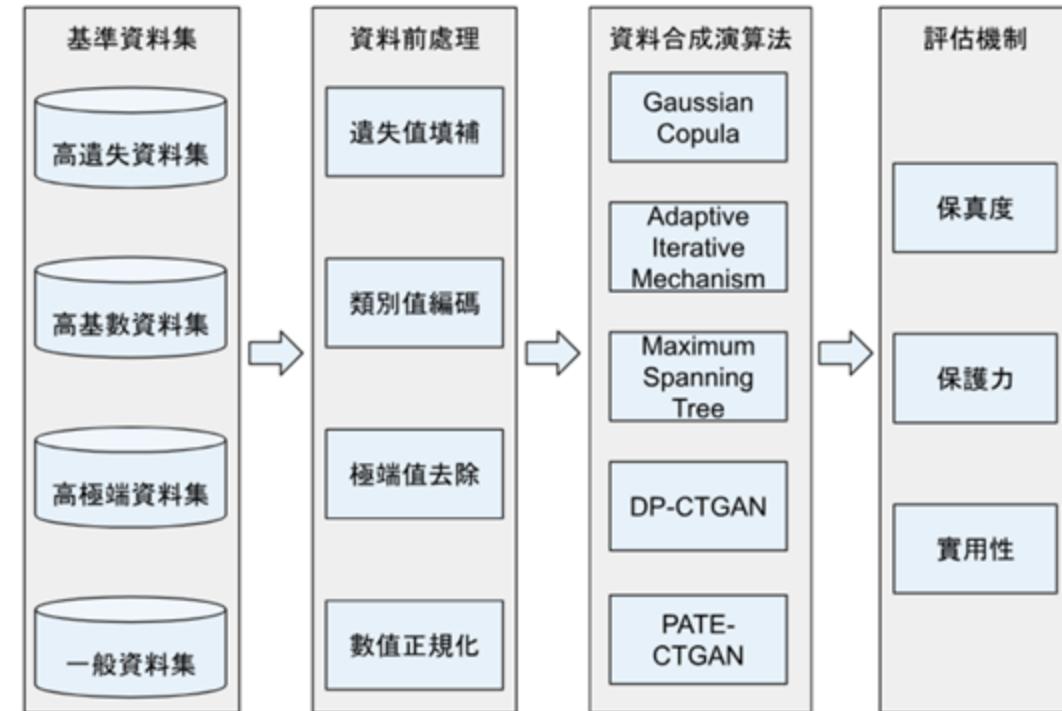
遺失值填補：填補

mean/median/mode或  
移除

類別值編碼：獨熱/均勻  
/標籤編碼

極端值去除：由 z-  
score/四分位距/隔離森  
林/局部離群因子來決定  
極端值

數值標準化：scale to  
[0,1], scale to [-1,1],  
standardize



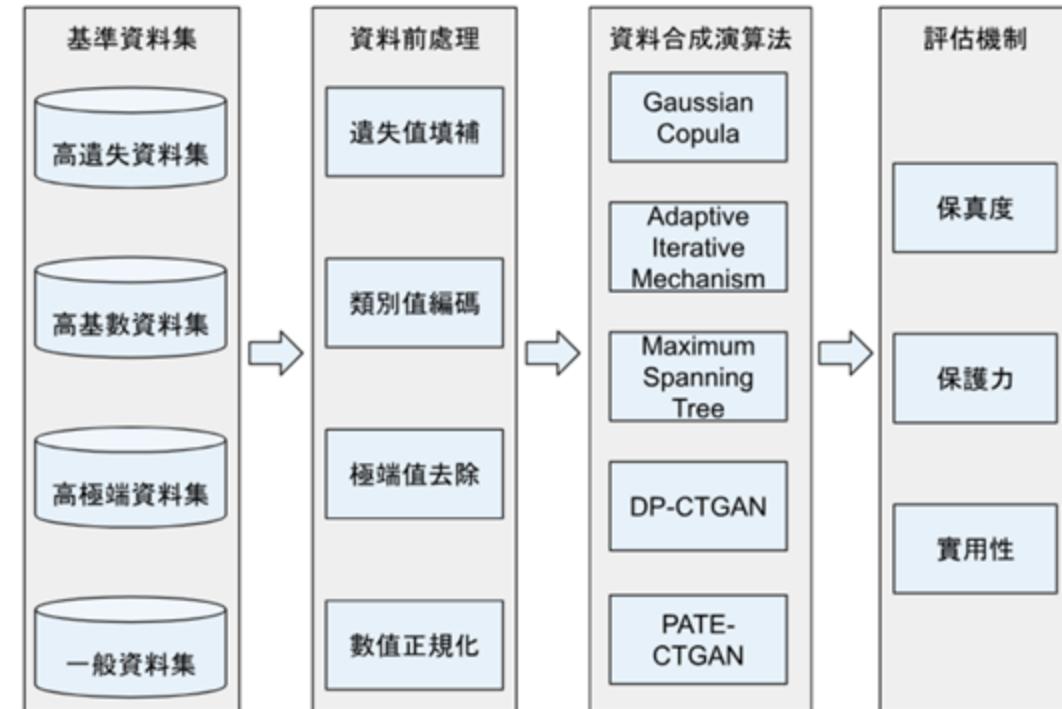
# 隱私強化技術演算法適用性分析 – 資料合成演算法

## 有 DP 保護

- Adaptive Iterative Mechanism (AIM), Maximum Spanning Tree (MST), DP-CTGAN, PATE-CTGAN

## 無 DP 保護

- Gaussian Copula



# 隱私強化技術演算法適用性分析 – 評估機制

保真度：合成資料是否能夠真實反映原始資料的特性

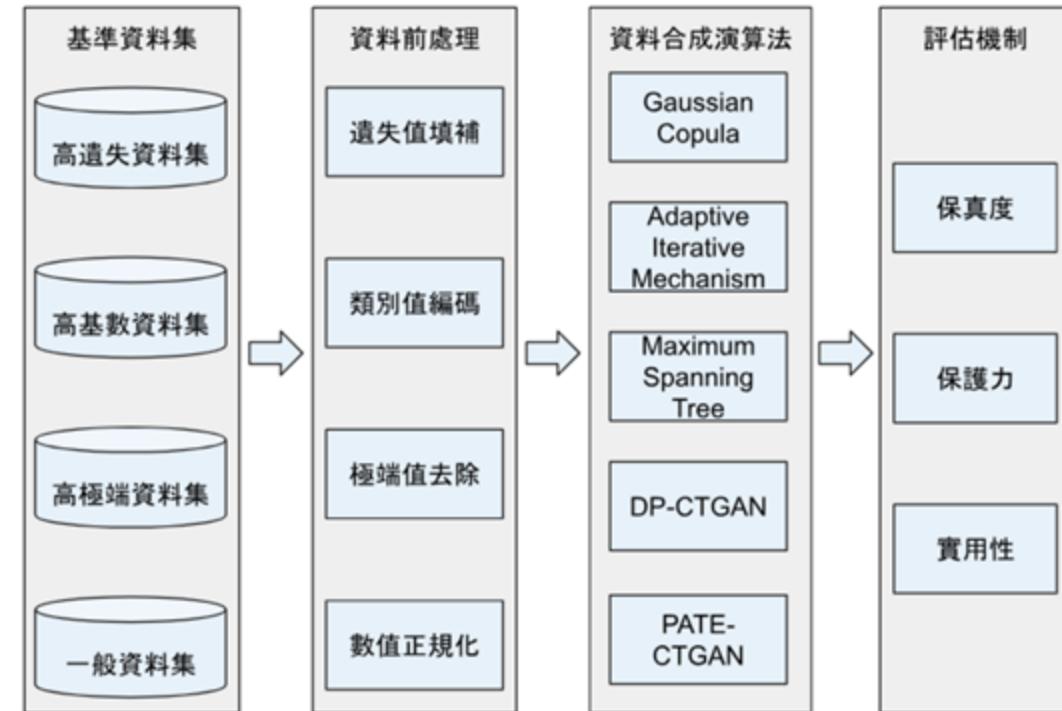
- 單欄統計特性、欄位關聯性等

保護力：合成資料承受攻擊的能力

- 指認性、連結性、推論性風險

實用性：合成資料與原始資料面對下游機器學習任務的成效差距

- 分類：Accuracy, AUROC, F1
- 迴歸：RMSE、R2
- 聚類：輪廓係數



# 隱私強化技術演算法適用性分析 – 前處理指引表

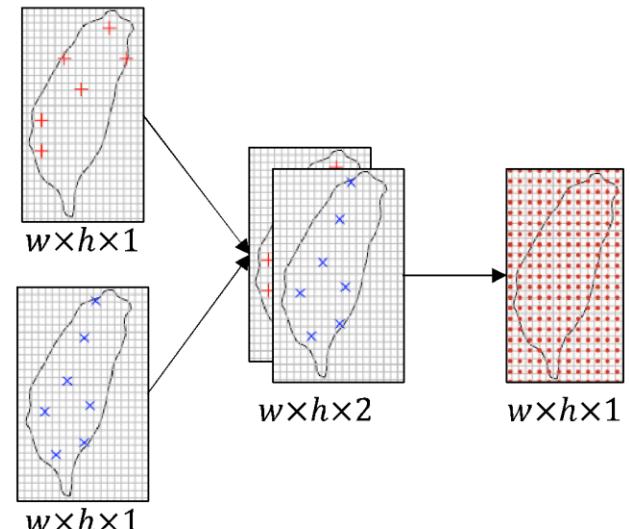
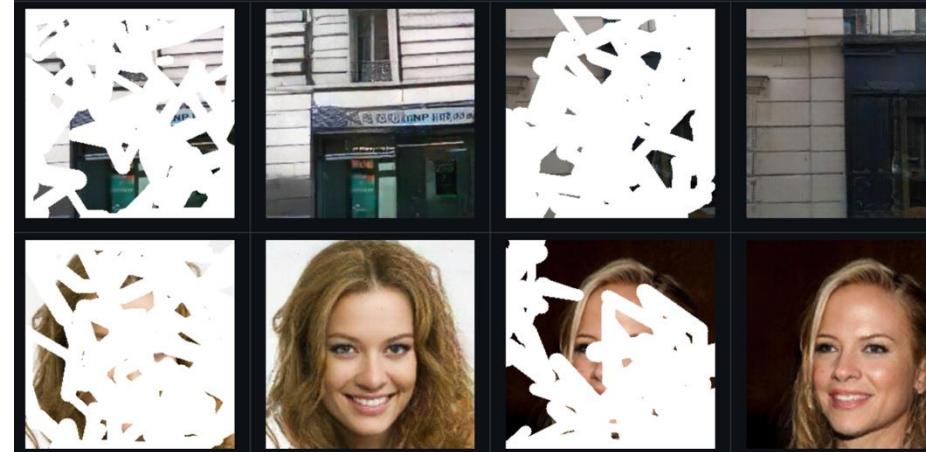
	重視保護力	重視保真度	重視實用性
面對高遺失資料集 (遺失值為數值)	使用 missing_median 處理遺失值	使用 missing_median 處理遺失值	使用 missing_median 處理遺失值
面對高遺失資料集 (遺失值為類別值)	使用 missing_mode 處理遺失值	使用 missing_drop 處理遺失值	使用 missing_mode 處理遺失值
面對高基數資料集	使用 encoder_label 對類別值編碼	使用 encoder_uniform 對類別值編碼	使用 encoder_label 對類別值編碼
面對高極端資料集	無明顯差異， 使用預設方法即可	使用 outlier_isolationforest 或 outlier_lof 處理極端值	使用 outlier_isolationforest 或 outlier_lof 處理極端值
面對一般資料集	無明顯差異， 使用預設方法即可	使用 scaler_minmax 對數值做標準化	無明顯差異， 使用預設方法即可

# 空汙感測網路異質資料整合 (2021 – 2023)

- 合作單位：中大大氣系、太空遙測中心、資工系
- 資工系負責項目：
  - 結合衛星資料與地表空汙感測器共同預測各地空汙

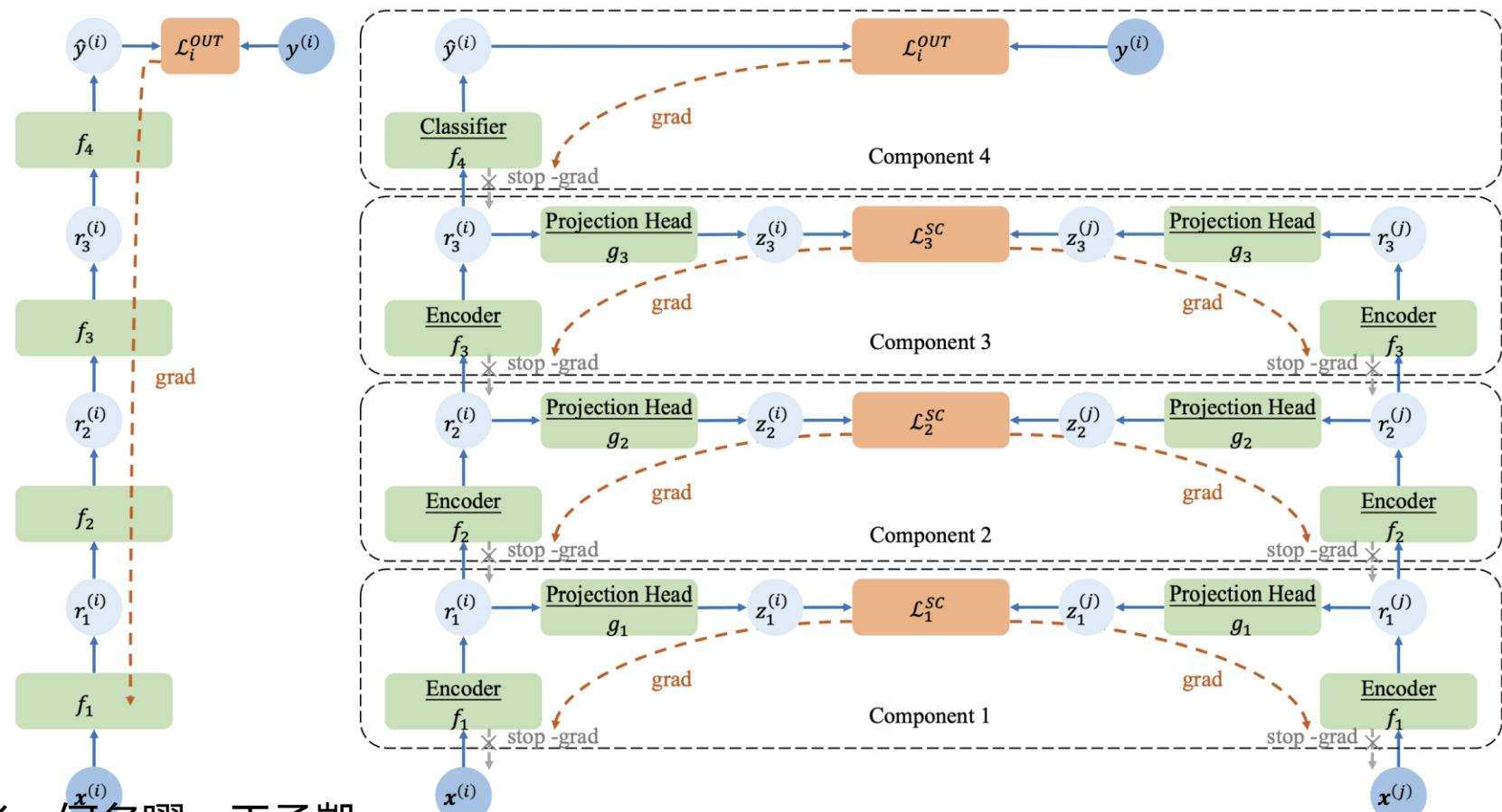
# 利用圖像修復技術進行多來源PM2.5的融合與補值

- 圖像修補技術
  - 從部份圖像生成整張圖像
  - 一張圖片有RGB三個頻道
- 多來源PM2.5融合
  - 從部份PM2.5測量生成全台PM2.5濃度地圖
  - 「EPA測站」、「Airbox測站」、及「衛星PM2.5」視為不同頻道
- 相同處
  - 均為三維陣列 (長  $\times$  寬  $\times$  頻道數)
  - 均需無中生有
- 相異處
  - 圖像修補任務中，相同(x,y)坐標點的不同頻道「同時有值」或「同時缺值」
  - PM2.5補值任務中，相同(x,y)坐標點的不同頻道不一定「同時有值」或「同時缺值」



# Supervised Contrastive Parallel Learning (SCPL) (1/3)

- Realizes model parallelism for deep learning; maintains high test accuracies across different networks and open datasets



# Supervised Contrastive Parallel Learning (SCPL) (2/3)

Standard BP

Device No.	Stage																
GPU0	FW1	FW2	FW3	FW4	LOSS	BW4	BW3		BW2				BW1			UP	
Time point	$t_1$	$t_2$	$t_3$	$t_4$	$t_5$	$t_6$	$t_7$	$t_8$	$t_9$	$t_{10}$	$t_{11}$	$t_{12}$	$t_{13}$	$t_{14}$	$t_{15}$	$t_{16}$	$t_{17}$

NMP

Device No.	Stage																
GPU0	FW1															UP	
GPU1		FW2														UP	
GPU2			FW3													UP	
GPU3				FW4	LOSS	BW4										UP	
Time point	$t_1$	$t_2$	$t_3$	$t_4$	$t_5$	$t_6$	$t_7$	$t_8$	$t_9$	$t_{10}$	$t_{11}$	$t_{12}$	$t_{13}$	$t_{14}$	$t_{15}$	$t_{16}$	$t_{17}$

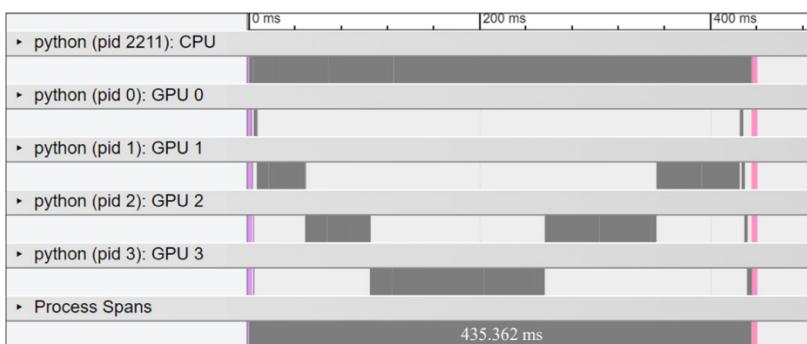
Concept illustration

SCPL

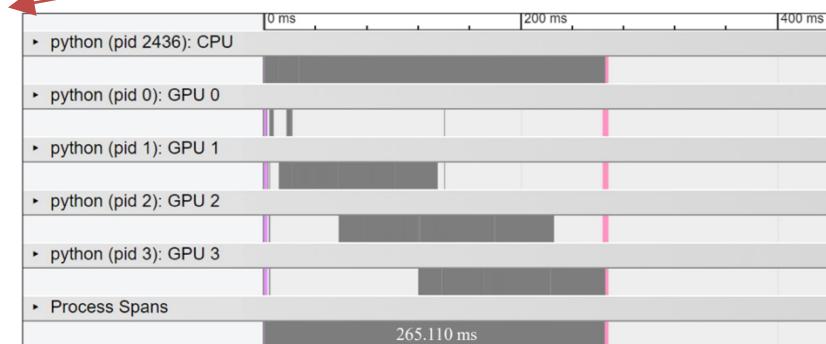
Device No.	Stage							
GPU0	FW1	LOSS	BW1				UP	
GPU1		FW2	LOSS	BW2				UP
GPU2			FW3	LOSS	BW3			UP
GPU3				FW4	LOSS	BW4		UP
Time point	$t_1$	$t_2$	$t_3$	$t_4$	$t_5$	$t_6$	$t_7$	$t_8$

FW $i$ : forward for layer  $i$   
 LOSS: compute loss  
 BW $i$  : backward for layer  $i$   
 UP: update parameter values

True training process on 4 GPUs



(a) Training LSTM on IMDB (using NMP).



(b) Training LSTM on IMDB (using SCPL).

# Supervised Contrastive Parallel Learning (SCPL) (3/3)

Training time speedup ratios (IMDB, transformer)

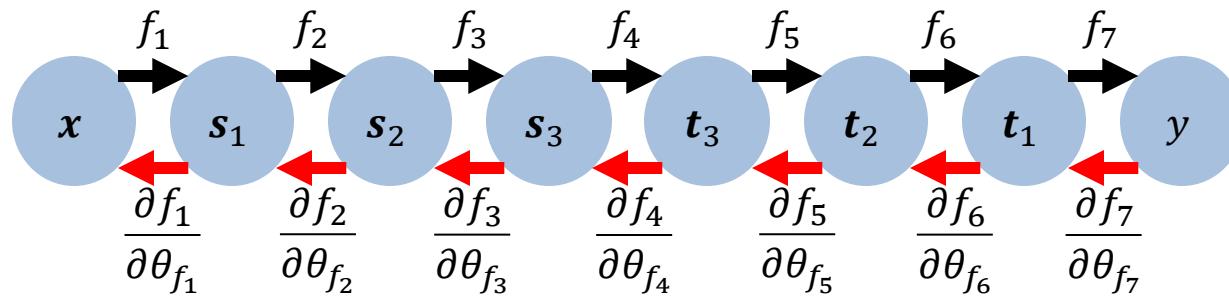
Batch size	32	64	128	256	512
BP	1x (196 min)	1x (173 min)	1x (156 min)	1x (149 min)	1x (147 min)
GPipe (1 GPU)	0.75x	0.72x	0.72x	0.71x	0.70x
GPipe (2 GPUs)	1.00x	0.92x	0.93x	0.93x	0.92x
GPipe (4 GPUs)	1.35x	1.25x	1.17x	1.16x	1.11x
SCPL (1 GPU)	1.12x	1.07x	1.03x	1.03x	1.05x
SCPL (2 GPUs)	1.43x	1.37x	1.32x	1.37x	1.38x
SCPL (4 GPUs)	1.92x	1.82x	1.66x	1.67x	1.66x

Test accuracies (IMDB)

		LSTM	Transformer
BP		89.68 $\pm$ 0.20	87.54 $\pm$ 0.44
Early Exit		84.34 $\pm$ 0.31	80.24 $\pm$ 0.24
AL		86.41 $\pm$ 0.61	85.65 $\pm$ 0.77
SCPL		<b>89.84 <math>\pm</math> 0.10 †</b>	<b>89.03 <math>\pm</math> 0.12 †</b>

# Associated learning (AL) (1/2)

- AL: an alternative to end-to-end backpropagation
- AL decomposes a network into small components:
  - Each component has a local objective function
  - Parameters in different components can be updated simultaneously
    - Eliminate backward lock, so pipelined training is possible; increase throughput



# Associated learning (2/2)

- Results on image classification (CIFAR-100)

	BP	AL
Vanilla CNN	$26.5 \pm 0.4\%$	<b><math>29.7 \pm 0.2\%</math></b>
VGG	$65.8 \pm 0.3\%$	<b><math>67.1 \pm 0.3\%</math></b>

- Results on NLP-1 (IMDB)

	BP	AL
LSTM	$88.10 \pm 0.50\%$	<b><math>89.04 \pm 0.37\%</math></b>

- Results on NLP-2 (AGNews)

	BP	AL
LSTM	$88.56 \pm 0.97\%$	<b><math>91.42 \pm 0.42\%</math></b>

# Parametric spectral clustering (1/2)

- Spectral Clustering
  - Pro: effective for non-linearly clustering
  - Con: high computational costs and memory usage
- Proposed Parametric Spectral Clustering (PSC)
  - Learns to project data into spectral space
  - Supports incremental clustering by updating clusters without retraining the entire model
  - Efficient for real-time scenarios with dynamic datasets

# Parametric spectral clustering (2/2)

- SC vs. PSC: computation cost and accuracy
  - Faster, memory-friendly, little (or no) sacrifice on accuracy

Method	Execution time (s)	Peak memory usage (MB)	ClusterAcc
SC	2462	5331	$0.794 \pm 0.04$
PSC ( $r = 1/6$ )	Training: 453 ( $\downarrow 82\%$ ) Inference: $0.443 \pm 0.027$	Training: 1032 ( $\downarrow 81\%$ ) Inference: $92.83 \pm 1.965$	$0.732 \pm 0.076$
PSC ( $r = 2/6$ )	Training: 533 ( $\downarrow 78\%$ ) Inference: $0.419 \pm 0.088$	Training: 1717 ( $\downarrow 68\%$ ) Inference: $92.63 \pm 2.09$	$0.739 \pm 0.089$
PSC ( $r = 3/6$ )	Training: 706 ( $\downarrow 71\%$ ) Inference: $0.437 \pm 0.048$	Training: 2532 ( $\downarrow 53\%$ ) Inference: $96.21 \pm 1.785$	$0.764 \pm 0.041$
PSC ( $r = 4/6$ )	Training: 1029 ( $\downarrow 58\%$ ) Inference: $0.379 \pm 0.06$	Training: 3328 ( $\downarrow 38\%$ ) Inference: $96.54 \pm 2.271$	$0.775 \pm 0.046$
PSC ( $r = 5/6$ )	Training: 1472 ( $\downarrow 40\%$ ) Inference: $0.491 \pm 0.036$	Training: 4937 ( $\downarrow 7\%$ ) Inference: $96.5 \pm 2.15$	$0.819 \pm 0.039$

# 偵測低調的網軍 (1/3)

- 電腦容易偵測高調的網軍
  - 常發言、常回文、常推/噓文等
- 偵測低調的網軍相對困難

AUPRC scores of detecting active and low active spammers

	active users	inactive users	diff
XGBoost	0.8892	0.5157	0.3735
LightGBM	0.7421	0.4888	0.2533
Random Forest	0.8317	0.5147	0.3163

- 但你知道大部份的網軍是「低調」的嗎？

Group	Percentile of active value	Active value	# normal accounts	CDF of normal accounts (a)	# spammers	CDF of spammers (b)	(b) – (a)
$G_1$	[0%, 10%)	0-18	4112	9%	222	24%	15%
$G_2$	[10%, 20%)	19-45	4418	20%	163	42%	22%
$G_3$	[20%, 30%)	46-84	4508	30%	86	52%	22%
$G_4$	[30%, 40%)	85-135	4223	40%	59	58%	18%
$G_5$	[40%, 50%)	136-211	4453	50%	57	64%	14%
$G_6$	[50%, 60%)	212-315	4096	59%	76	73%	14%
$G_7$	[60%, 70%)	316-494	4320	69%	112	85%	16%
$G_8$	[70%, 80%)	495-817	4368	79%	67	92%	13%
$G_9$	[80%, 90%)	818-1663	4638	90%	51	98%	8%
$G_{10}$	[90%, 100%]	$\geq 1664$	4554	100%	19	100%	0%

# 偵測低調的網軍 (2/3)

- 使用傳統機器學習或深度學習偵測低活躍網軍成效不彰

AUPRC scores of detecting less active and highly active spammers

	[0%, 10%)	[10%, 20%)	[80%, 100%)
XGBoost	0.52 ± 0.01	0.48 ± 0.03	0.89 ± 0.01
LightGBM	0.49 ± 0.02	0.40 ± 0.04	0.74 ± 0.02
Random Forest	0.51 ± 0.03	0.27 ± 0.02	0.83 ± 0.02
Fully Connected	0.35 ± 0.06	0.38 ± 0.05	0.75 ± 0.03
ConvNet	0.17 ± 0.06	0.26 ± 0.14	0.80 ± 0.33
Soft Voting [22]	0.40 ± 0.01	0.43 ± 0.01	0.76 ± 0.01
Hard Voting [22]	0.43 ± 0.02	0.47 ± 0.02	0.70 ± 0.03
Stacking [22]	0.42 ± 0.01	0.47 ± 0.03	0.67 ± 0.01

- GNN模型 vs. 最佳非 GNN 模型：GNN更精確地偵測低活躍網軍

GNN vs. XGBoost (best among non-GNN models)

	[0%, 10%)	[10%, 20%)	[80%, 100%)
XGBoost	0.52 ± 0.01	0.48 ± 0.03	0.89 ± 0.01 †
GCN	<b>0.66</b> ± 0.18	0.38 ± 0.13	0.72 ± 0.07
TAGCN ( $K = 1$ )	<b>0.64</b> ± 0.04	<b>0.79</b> ± 0.06	<b>0.89</b> ± 0.07 †
TAGCN ( $K = 2$ )	<b>0.68</b> ± 0.02	<b>0.84</b> ± 0.05 †	<b>0.89</b> ± 0.08 †
TAGCN ( $K = 3$ )	<b>0.71</b> ± 0.04 †	<b>0.80</b> ± 0.07	<b>0.89</b> ± 0.06 †
GAT	<b>0.62</b> ± 0.09	<b>0.77</b> ± 0.05	<b>0.89</b> ± 0.06 †

# 偵測低調的網軍 (3/3)

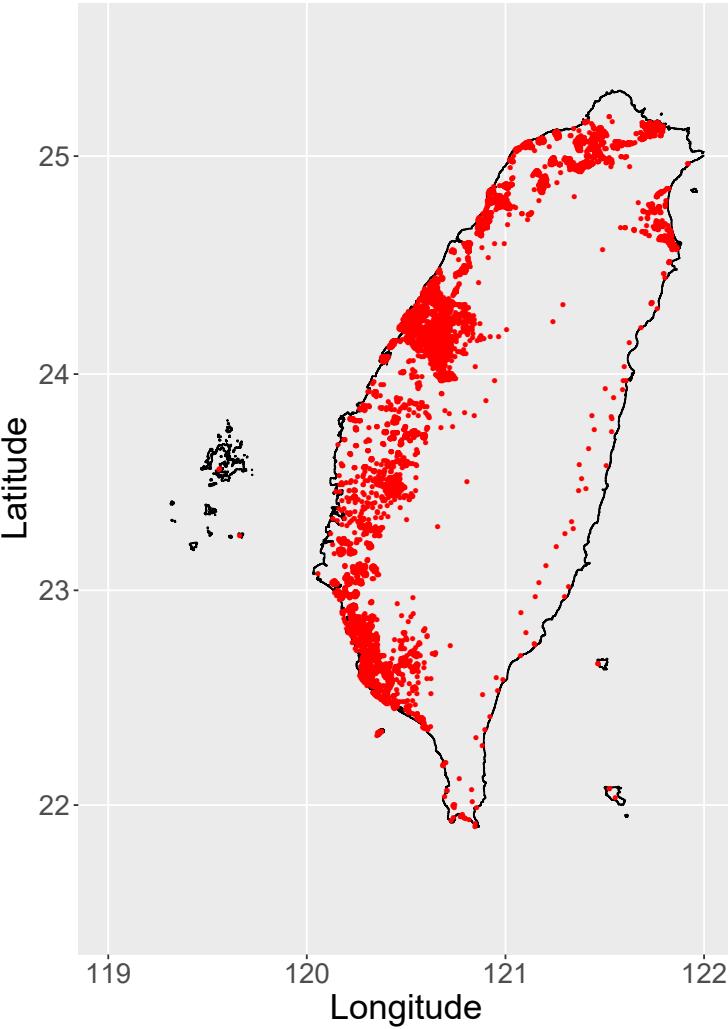
- 加入社群特徵可幫助所有模型更好地偵測網軍

AUPRC scores of the models when including social features

Type	Model	[0%, 10%)	[10%, 20%)	[80%, 100%)	[0%, 100%)
Non-GNN-based models (including social features)	XGBoost	$0.83 \pm 0.01$	<b>0.74</b> $\pm 0.03$	<b>0.90</b> $\pm 0.02$	<b>0.86</b> $\pm 0.00$
	LightGBM	<b>0.86</b> $\pm 0.02$	$0.72 \pm 0.05$	$0.88 \pm 0.02$	$0.82 \pm 0.00$
	Random Forest	$0.85 \pm 0.01$	$0.56 \pm 0.05$	$0.85 \pm 0.02$	$0.79 \pm 0.00$
	Fully Connected	$0.53 \pm 0.07$	$0.51 \pm 0.06$	$0.76 \pm 0.05$	$0.64 \pm 0.04$
	ConvNet	$0.43 \pm 0.09$	$0.68 \pm 0.07$	$0.83 \pm 0.04$	$0.66 \pm 0.06$
	Soft Voting [22]	$0.69 \pm 0.00$	$0.56 \pm 0.01$	$0.76 \pm 0.01$	$0.72 \pm 0.00$
	Hard Voting [22]	$0.67 \pm 0.01$	$0.63 \pm 0.02$	$0.70 \pm 0.03$	$0.74 \pm 0.01$
	Stacking [22]	$0.54 \pm 0.02$	$0.56 \pm 0.03$	$0.67 \pm 0.01$	$0.69 \pm 0.02$
GNN-based models (including social features)	GCN	$0.62 \pm 0.08$	$0.52 \pm 0.05$	$0.83 \pm 0.08$	$0.69 \pm 0.03$
	TAGCN ( $K = 1$ )	$0.79 \pm 0.03$	<b>0.97</b> $\pm 0.05$	<b>0.99</b> $\pm 0.04$	<b>0.92</b> $\pm 0.01$
	TAGCN ( $K = 2$ )	$0.82 \pm 0.03$	<b>0.98</b> $\pm 0.02$	<b>0.99</b> $\pm 0.03$	<b>0.93</b> $\pm 0.02$
	TAGCN ( $K = 3$ )	$0.85 \pm 0.02$	<b>0.98</b> $\pm 0.03$	<b>0.98</b> $\pm 0.01$	<b>0.94</b> $\pm 0.01$
	GAT	$0.73 \pm 0.06$	<b>0.91</b> $\pm 0.06$	<b>0.92</b> $\pm 0.07$	<b>0.87</b> $\pm 0.05$

# 空汙感測器故障預測 – supervised learning-based

- 10,000+ 空汙感測器 (in 2021), 但有相當比例之量測值不精準
- 採定期巡檢, 但人力成本極高
- 智慧巡檢: 以圖卷積網路 (Graphical Convolutional Network) 與時間卷積網路整合時空資訊預測故障之感測器
- 訓練資料採用 2018 年的部份資料
- 工研院於 2018 年 5 月至 12 月巡檢 144 個測站, 以巡檢結果做為測試資料
  - 28 個異常
  - 116 個正常
  - 我們以此巡檢紀錄評估各種異常偵測演算法的優劣



# 實驗結果 – AUROC 分數

Type	Model	ROC mean	ROC std
Rule based	ADF-5 (5 是 [6] 中給的超參數值)	0.624	0.0
	ADF-10(ROC Best)	0.694	0.0
ML(無圖卷積)	Random Forest	0.6878	0.006261
	Lasso	0.7000	0.015652
	Ridge	0.7085	0.013472
	TCN	0.7066	0.007701
	DNN	0.6940	0.007211
	LSTM	0.7090	0.007211
ML(圖卷積)	GraphWaveNet	<b>0.7260</b>	<b>0.010826</b>
	STGCN	0.7214	0.018569

# 實驗結果 – Precision@ $k$

- Precision@ $k$ : 若按建議依序檢查 $k$ 個測站，  
實際有問題的測站在 $k$ 個測站中的佔比

Type	Model	P@10	P@20	P@30	P@40	P@50
隨機巡檢		0.194	0.194	0.194	0.194	0.194
Rule based	ADF-5	0.300	0.350	0.270	0.330	0.320
	ADF-10(ROC Best)	0.500	0.500	0.400	0.380	0.320
ML(無圖卷積)	Random Forest	0.380	0.370	0.400	0.342	0.320
	Lasso	0.580	0.430	0.394	0.370	0.320
	Ridge	0.600	0.433	0.395	0.375	0.337
	TCN	0.600	0.410	0.412	0.338	0.320
	DNN	0.500	0.430	0.374	0.344	0.312
	LSTM	0.600	0.410	0.368	0.332	0.336
ML(圖卷積)	GraphWaveNet	0.600	0.417	<b>0.417</b>	0.380	0.353
	STGCN	<b>0.640</b>	<b>0.450</b>	0.398	<b>0.386</b>	<b>0.360</b>

# 實驗結果 – Recall@ $k$

- Recall@ $k$ : 按建議依序檢查 $k$ 個測站，找出有問題的測站數量與實際有問題測站數量(28個)的比值

Type	Model	R@10	R@20	R@30	R@40	R@50
隨機巡檢		0.069	0.139	0.208	0.278	0.347
Rule based	ADF-5	0.110	0.250	0.290	0.460	0.570
	ADF-10(ROC Best)	0.180	0.360	0.430	0.540	0.570
ML(無圖卷積)	Random Forest	0.136	0.266	0.428	0.484	0.570
	Lasso	0.204	0.306	0.422	0.524	0.570
	Ridge	0.210	0.308	0.423	0.533	0.603
	TCN	0.212	0.296	0.442	0.476	0.570
	DNN	0.180	0.308	0.398	0.484	0.560
	LSTM	0.214	0.293	0.394	0.474	0.600
ML(圖卷積)	GraphWaveNet	0.214	0.300	<b>0.447</b>	0.543	0.630
	STGCN	<b>0.230</b>	<b>0.322</b>	0.428	<b>0.550</b>	<b>0.642</b>

# 空汙感測器故障預測 – semi-supervised learning-based

- 10000+ 個空汙感測器中，只有144個有「正常」或「故障」的標準答案
- Fully supervised learning: 僅有 144 筆訓練資料
- Semi-supervised learning: 融合有標準答案的資料及其他沒有標準答案的資料共同訓練

# 空汙感測器故障預測 – semi-supervised learning-based

		隨機巡檢	0.1940 ± 0.0000		
非機器學習模型	ADF-5		0.2900 ± 0.0000		
	ADF-10		0.4400 ± 0.0000		
		折線圖	熱力圖	統整性資料	統整及時序資料
監督式模型	linear regression	0.2769	0.3137	0.3339	0.3163
	ridge regression	0.3214	0.3876	0.3337	0.3159
	random forest	0.3290	0.4292	<b>0.4471</b>	<b>0.4588</b>
	SSDO with iforest	0.3374	0.4555	0.3061	0.2883
	SSDO with COP-kmeans	0.3399	0.5158	0.3177	0.2554
無監督式模型	Isolation fores	0.1886	0.2003	0.2375	0.2578
半監督式模型	SSDO with iforest	0.3712	0.4114	0.2645	0.3773
	SSDO with COP-kmeans	0.3640	0.4162	0.2809	0.3214
	Deep SAD	<b>0.8099</b>	<b>0.8048</b>	0.3450	0.4215



不同模型在不同資料中所得到的PR-AUC

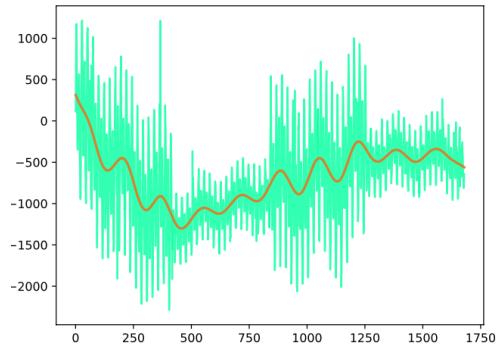
# Extended Clickstream

- Weblog approximately records only half of a user's page visits
- 8.1% of the visits recorded in the weblog may not come from a user's conscious actions
- Clickstream is an incomplete collection of users' web visiting

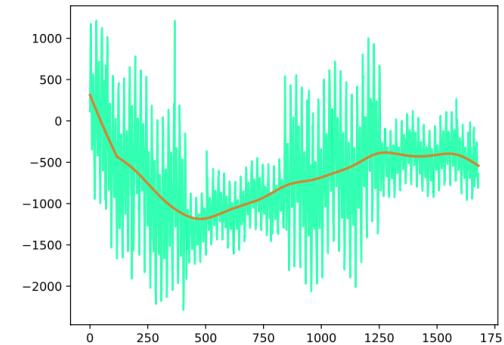
Category	ICS + ECS				CS		ICS		ECS		Rank Diff (1)-(2)
	Rank(1)	Count	Perc. (%)	CDF (%)	Rank(2)	Count	Rank	Count	Rank	Count	
Streaming Media and Download	1	1110256	17.57	17.57	3	558327	2	541878	1	568378	-2
Social Networking	2	929709	14.72	32.29	1	608252	1	591064	2	338645	1
Search Engines and Portals	3	709671	11.23	43.52	2	559254	3	456281	5	253390	1
Education	4	558183	8.84	52.36	5	304386	5	263847	3	294336	-1
Information Technology	5	449954	7.12	59.48	6	200185	6	181300	4	268654	-1
Web-based Applications	6	390278	6.18	65.66	4	336890	4	331990	11	58288	2
Games	7	379462	6.01	71.67	7	156351	7	145209	6	234253	0
Business	8	199455	3.16	74.83	9	108063	10	95567	7	103888	-1
Shopping	9	166820	2.64	77.47	11	94739	11	86591	8	80229	-2
File Sharing and Storage	10	163682	2.59	80.06	10	106536	9	102926	10	60756	0
Entertainment	11	153140	2.42	82.48	8	117183	8	113604	14	39536	3
Reference	12	152565	2.41	84.89	12	86090	12	78747	9	73818	0
Web-based Email	13	113965	1.8	86.69	13	68743	13	66548	12	47417	0
News and Media	14	99934	1.58	88.27	14	67278	14	65898	17	34036	0
Newsgroups and Message Boards	15	71043	1.12	89.39	16	35037	17	31629	15	39414	-1
Pornography	16	68720	1.09	90.48	15	42031	15	39897	18	28823	1
Personal Websites and Blogs	17	68312	1.08	91.56	20	25497	20	24055	13	44257	-3
Instant Messaging	18	62816	0.99	92.55	18	29973	18	28458	16	34358	0
Auction	19	55353	0.88	93.43	17	33344	16	32504	20	22849	2
Travel	20	48802	0.77	94.2	19	29955	19	24893	19	23909	1

# 個人化之趨勢線生成 (1/2)

- 哪條才是趨勢線?
  - 不同情境, 不同答案
  - 有人希望趨勢線「平滑」, 有人希望趨勢線仍能有「局部起伏」
- 如何讓電腦「學習」一個人心中的趨勢線樣貌? --  
個人化趨勢線生成
- Training: 系統展示十張時間序列, 使用者分別標注其心目中的趨線, 系統從中學習使用者想要的趨勢線樣貌
- Generation: 使用者給予系統所有需要標示趨勢線之時間序列, 系統按 training 時學習到之規則自動為所有時間序列標出趨勢線
  - 挑戰: 僅有十張訓練資料, 如何有效的學習 (且不 overfitting)



(a) Trend line 1



(b) Trend line 2

# 個人化之趨勢線生成 (2/2)

- 兩階段之個人化趨勢線生成技術
- DNN model 容易 overfitting
- Pretrain and finetune 有部份效果，但仍不理想
- Petrel (我們的方法) 優於上面兩類

Type	Algorithm	SMAPE	MSE	Algorithm	SMAPE	MSE
Our method	Petrel (averaged)	<b>0.44</b>	5264.34	Petrel (averaged)	0.33	6164.38
	Petrel (weighted)	<b>0.44</b>	<b>5258.34</b>	Petrel (weighted)	<b>0.32</b>	<b>6002.32</b>
DNN models	ConvNet	0.83	176593.87	ConvNet	0.94	166951.8
	LSTM	1.02	497312.33	LSTM	1.11	323712.95
	Transformer	1.08	579188.89	Transformer	1.20	637955.96
DNN with pretraining and fine-tuning	P&F ConvNet	<b>0.44</b>	5425.77	P&F ConvNet	1.45	241890.91
	P&F LSTM	0.52	7394.09	P&F LSTM	1.23	1292454.44
	P&F Transformer	0.47	9311.75	P&F Transformer	0.81	1357013.58
	P&F MLP	0.68	31934.92	P&F MLP	1.18	242234.14

資料集一

資料集二

# E-commerce object and behavior embedding (Behavior2Vec)

- Predict a user's next clicked item
- Predict a user's next purchased item
- Discover the relationship between items
  - E.g., Canon's camera body : Canon's lens  $\approx$  Nikon's camera body : Nikon's lens

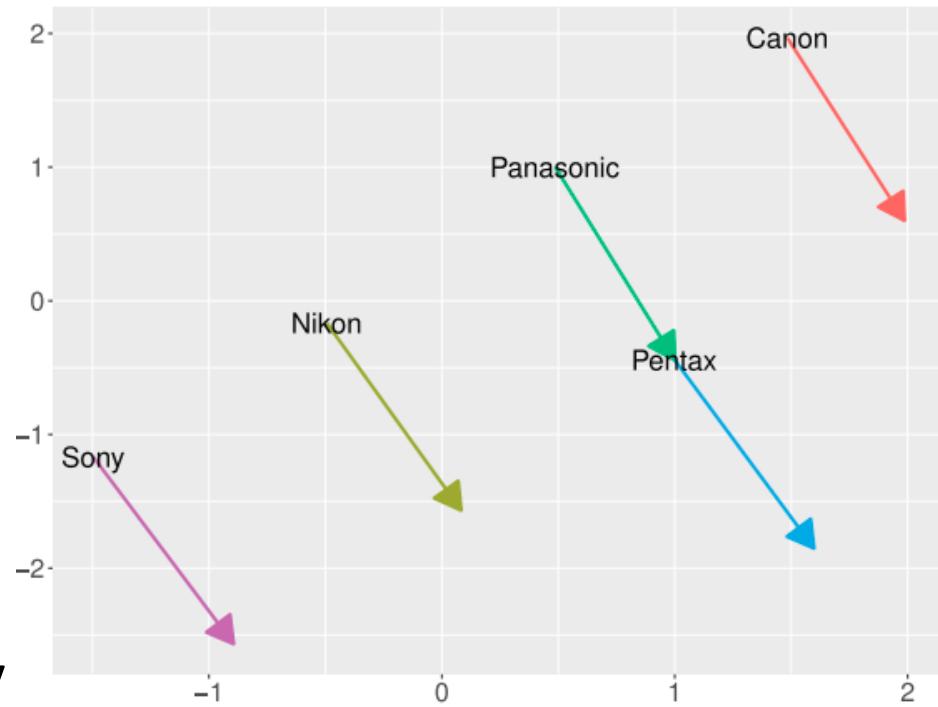


Figure 1: Vectors from the camera body to the corresponding kit lens of different brands. The vectors are generated by Behavior2Vec

# Recommendation for near cold start items

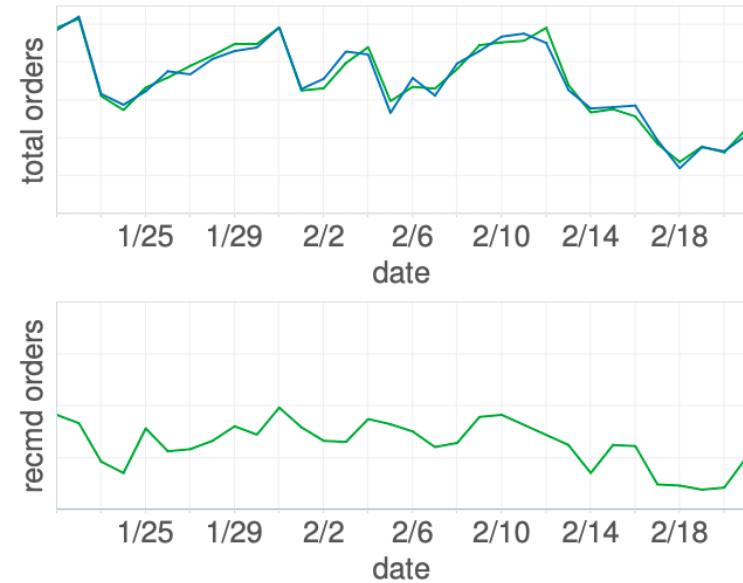
- Near cold start item: items that are rarely viewed
- Recommendation for the near cold start items is difficult because of the limited clues
- Our RDF method alleviates this issue

Table 1: a comparison of the methods with RDF and without RDF

Dataset	SVD	linear-reg	sqrt-reg	log-reg	improve ratio range
Epinions	1.1997	<b>1.0538</b>	<b>1.0538</b>	<b>1.0538</b>	12.16%
MovieLens-100K	0.9423	<b>0.9422</b>	<b>0.9422</b>	<b>0.9422</b>	0.01%
FilmTrust	0.8465	<b>0.8194</b>	<b>0.8194</b>	0.8223	2.86% to 3.20%
Yahoo! Movies	3.0799	<b>2.9892</b>	3.0129	3.0127	2.18% to 2.94%
AMI	1.1450	<b>1.1405</b>	<b>1.1405</b>	<b>1.1405</b>	0.39%

# Train and evaluate recommender systems in the right way

- Show 4 common errors in training and evaluating recommender systems
- Propose solutions or work-arounds for these issues



Green: channel with a recommendation  
Blue: channel w/o recommendation

# Co-learning user's browsing tendency of multiple categories

- Instead of predicting each target variable independently, our MFMT method simultaneously learns multiple targets in one model

Table: F1 scores of different models on different target categories

model	shopping	traveling	restaurant and dining	entertainment	games	education	
kNN	0.574	<b>0.615</b>		0.528	0.440	0.492	0.484
Logreg	0.578	0.489		0.501	0.402	0.441	0.437
SVM	0.576	0.391		0.410	0.399	0.409	0.385
MFMT	<b>0.584</b>	0.570		<b>0.561</b>	<b>0.479</b>	<b>0.531</b>	<b>0.515</b>
	(win)			(win)	(win)	(win)	(win)

# User personality and demographic profile prediction based on browsing logs

Table: errors of the personality test score prediction based on the supervised learners with and without the preprocessing step

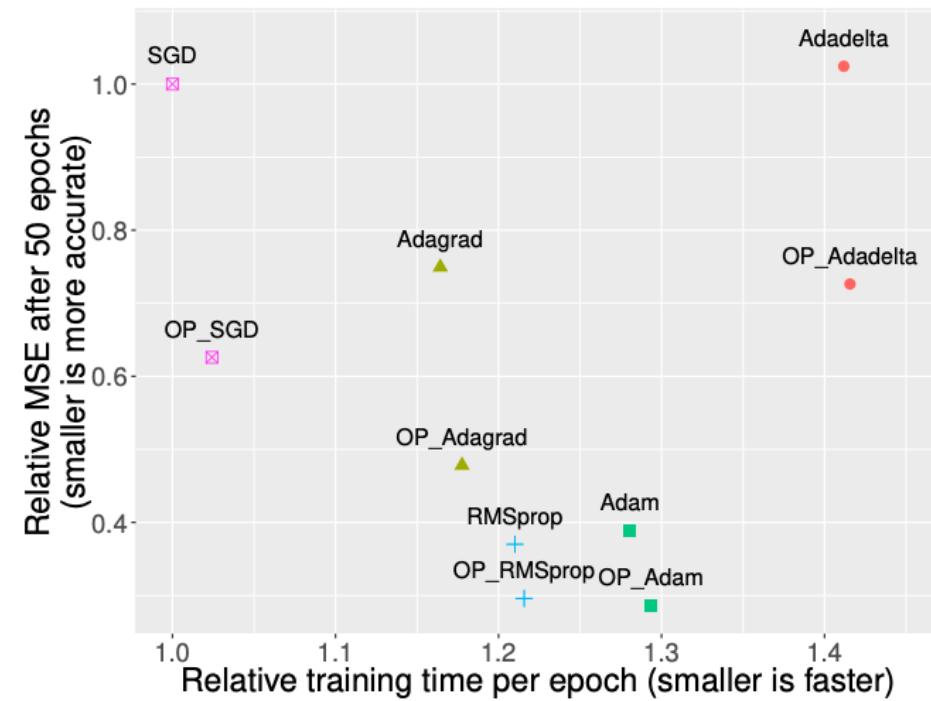
Method	Supervised regressor						Clustering + supervised regressor (win)					
Prediction target	HH	Neu	Ext	Agr	Con	Ope	HH	Neu	Ext	Agr	Con	Ope
Lasso	5.832	5.87	5.881	5.71	5.406	5.607	5.411	5.469	5.435	5.435	5.022	5.131
Ridge	5.845	5.981	5.891	5.795	5.43	5.646	5.43	5.404	5.38	5.325	5.027	5.052
Elastic net	5.813	5.769	5.743	5.622	5.366	5.44	5.417	5.383	5.422	5.317	5.022	5.095
SVR	5.789	5.78	5.746	5.643	5.232	5.38	5.432	5.623	5.402	5.328	5.048	5.165

Table: MicroF1 scores of the demographical info prediction based on the supervised learners with and without the preprocessing step

Method	Supervised classifier			Clustering + supervised classifier			(mostly win)
Prediction target	Age	Gender	Relationship	Age	Gender	Relationship	
Baseline	0.388	0.545	0.474	0.411	0.598	0.476	
KNN	0.427	0.594	0.478	0.435	0.618	0.482	
Random Forest	0.453	0.697	0.488	0.419	0.687	0.512	
Logistic Regression	0.427	0.697	0.476	0.457	0.675	0.498	
SVM	0.388	0.591	0.474	0.411	0.642	0.512	

# Accelerating MF by Overparameterization

- Overparameterization significantly accelerates the optimization of MF
  - Theoretically derive that applying the vanilla SGD on OP\_MF is equivalent to using GD with momentum and adaptive learning rate on the standard MF model



# Public transportation optimization

- Predict the taxi demand in real time by deep learning

Model	RMSE	MAPE
Average	$8.845 \pm 7.9434$	$0.0840 \pm 0.000413$
ARIMA	$15.585 \pm 20.8253$	$0.1660 \pm 0.018033$
ridge regression	$10.914 \pm 2.4451$	$0.1460 \pm 0.000895$
XGBoost	$6.498 \pm 2.0542$	$0.0806 \pm 0.000205$
LSTM (2 layers)	$7.037 \pm 3.9747$	$0.0563 \pm 0.000056$
LSTM (4 layers)	$6.694 \pm 5.1110$	$0.0595 \pm 0.000232$
DMVST-Net	$7.350 \pm 3.7034$	$0.0643 \pm 0.000192$
<b>ResLSTM (4 layers)</b>	<b><math>5.187 \pm 2.0265</math></b>	<b><math>0.0584 \pm 0.000048</math></b>
<b>AR-LSTM (4 layers)</b>	<b><math>4.958 \pm 1.8909</math></b>	<b><math>0.0488 \pm 0.000039</math></b>

# Dynamic ensembled learning

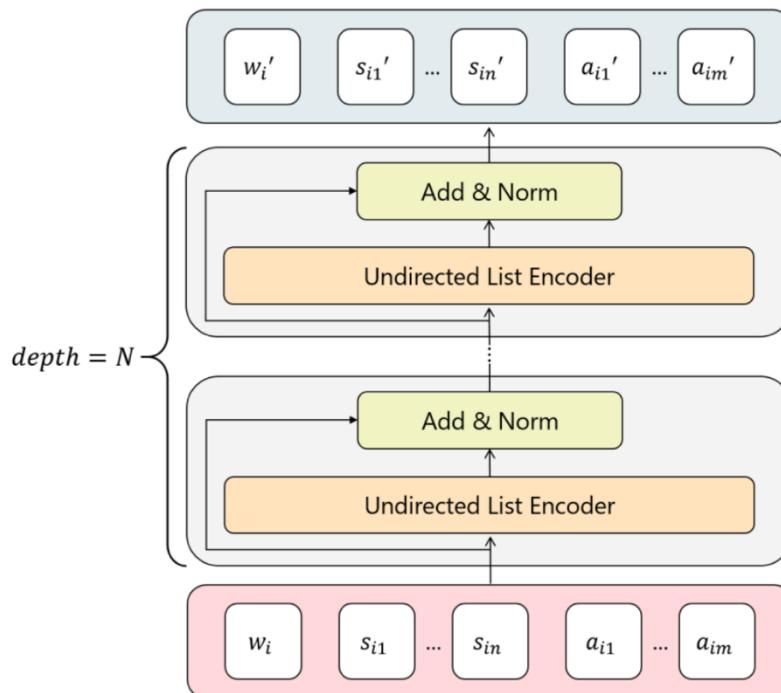
- Dynamically integrate multiple base learners based on the feature distribution of the test instance
- Better accuracy than the static ensembled learning approach

Table: a comparison of the base learners, static ensembled, and dynamic ensembled methods

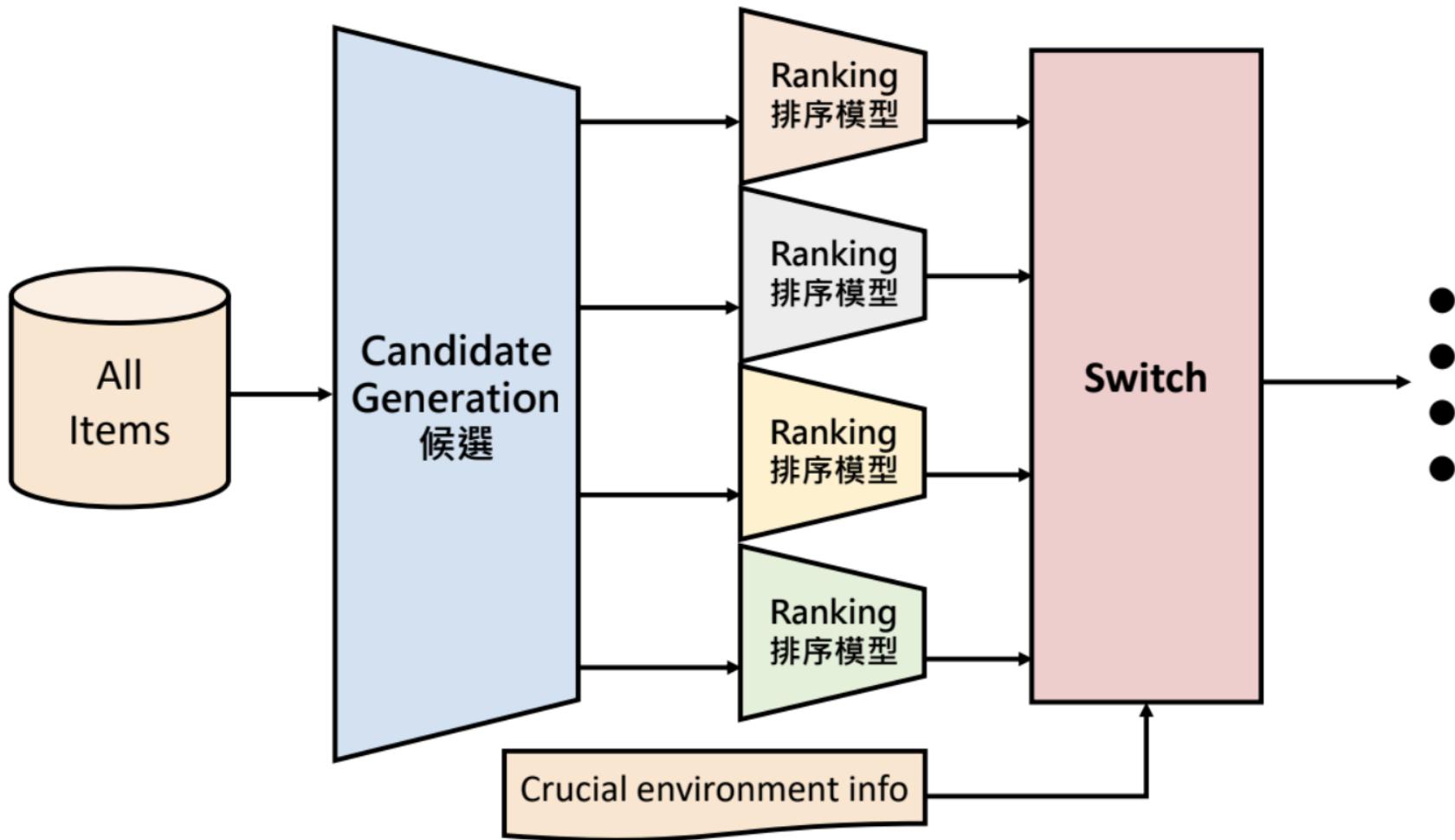
Method	KNN	SVM	Decision Tree	Majority Voting	Dynamic ensembled
Accuracy	77.09%	72.77%	75.46%	77.64%	77.80% (win)

# Better word embedding for synonyms and antonyms

- Adjusting word embedding to differentiate synonyms and antonyms



# Deep vs shallow recommendation



# Recent undergraduate projects (大學專題)

# 抗旋轉之卷積網路設計

- 卷積網路難以判斷旋轉的圖片
  - 從電腦的角度來看，旋轉前後的圖片之pixel排列方式不同，故可能認定圖片中為不同的物件
- 深度學習通常需要讓電腦看過各種旋轉角度的圖片，讓電腦「認得」不同旋轉角度的相同物件
- 我們設計新的模型，電腦只需看過一張圖，即可認得各種旋轉角度的圖片



Test accuracy

	MNIST		FashionMNIST		CIFAR-10	
	轉90度	任意旋轉	轉90度	任意旋轉	轉90度	任意旋轉
ConvNet1	0.17	0.42	0.07	0.22	0.29	<b>0.30</b>
ConvNet2	0.16	0.33	0.02	0.19	0.24	0.23
Our model	<b>0.72</b>	<b>0.43</b>	<b>0.79</b>	<b>0.33</b>	<b>0.36</b>	0.26

# 學術搜尋引擎/關鍵字標註器

- Build an academic search engine for the Taiwanese Association for Artificial Intelligence (中華民國人工智慧學會)
  - <http://search.taaai.org.tw/>
- Keyword search
- Paper keyphrase extractor

中華民國人工智慧學會 Taiwanese Association for Artificial Intelligence

學術搜索

請輸入關鍵字

keywords...

Hottest Top 10 Paper

多層式短中長期記憶模型之即時計程車需求預測

2018 TaaI Domestic track

作者: 徐志榮, 陳弘軒

點擊: 11 次

摘要 — 智慧交通儼然成為智慧城市的重要一環，運用人工智慧科技進行計程車需求預測是其中一項課題。有效地預測下個時間點載客需求的分布可以減少司機空車時間、降低乘客等待時間及增加獲利載客次數，將計程車產業獲利最大化並解決車輛巡迴載客所造成的能源消耗及汙染。本文利用計程車行車紀錄資料結合深度學習的架構 提出有效的計程車載客需求預測模型，使用善於處理時間序列架構的短中長期記憶模型(LST...

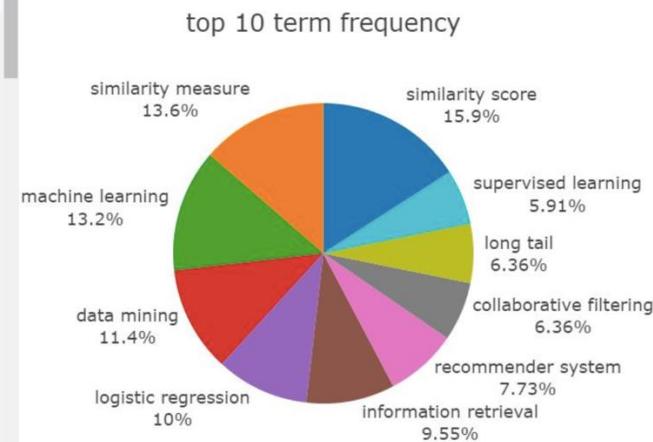
Multi-dependent Latent Dirichlet Allocation

2017 TaaI International track

作者: WEI CHENG HSIN, Jen-Wei Huang

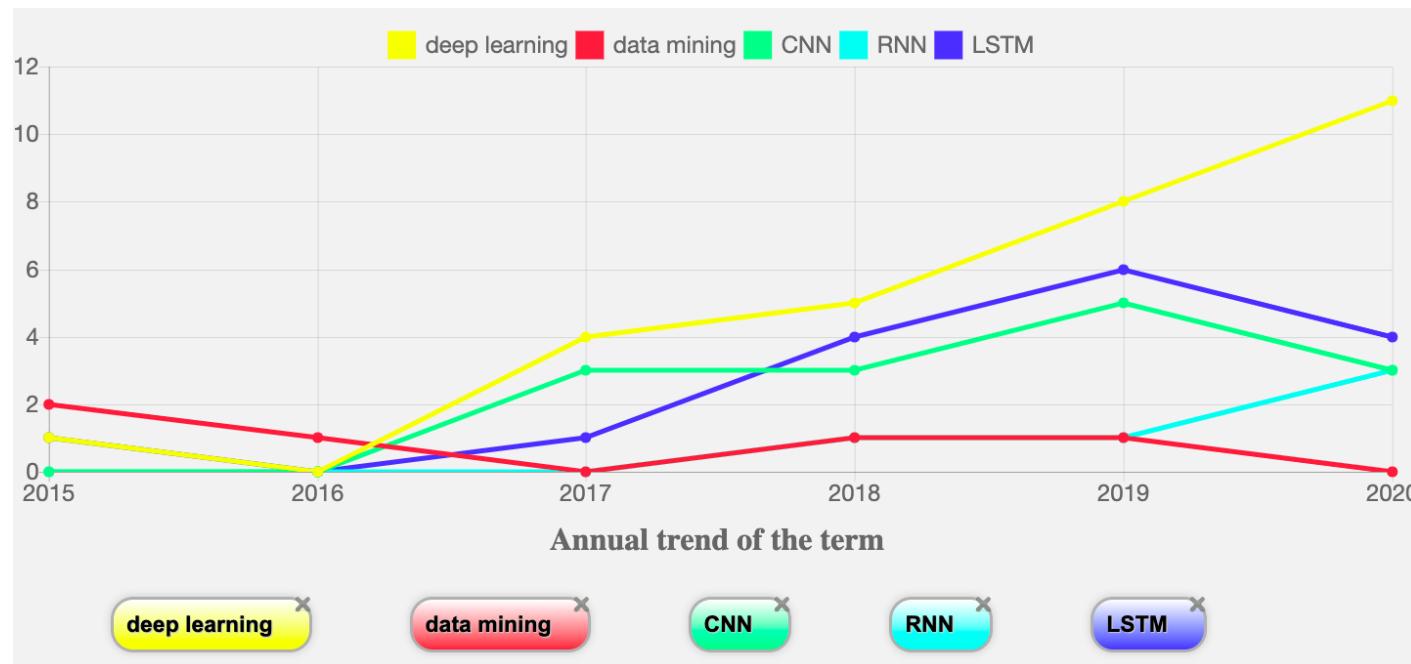
點擊: 6 次

Latent Dirichlet Allocation (LDA) is an attractive topic model research because LDA is so flexible for solving different problems. Because of its different core dependencies, it can be applied to many topics, such as emotion detection, information systems or image clustering. In recent works,



# 基於學術搜尋引擎之研究趨勢分析

- 利用 TAAI 論文發表內容分析研究主題歷年趨勢



# Instagram騷擾帳號偵測

## 初始畫面

Fake Account Detector

Please enter a username:

Submit

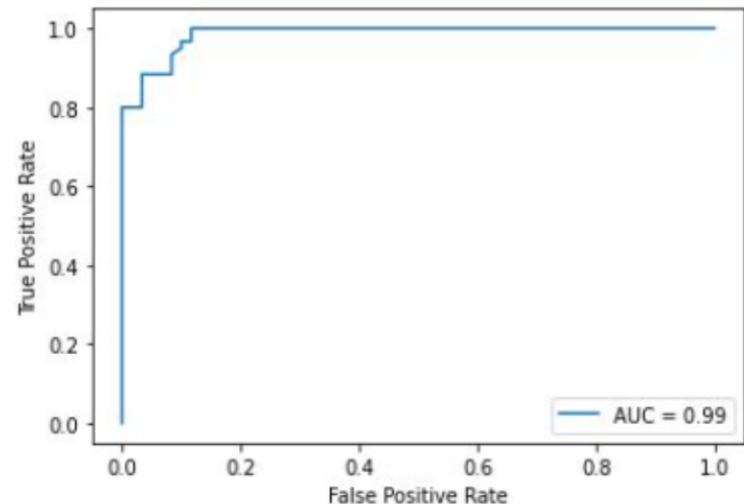
**⚠ This might be a FAKE account!**

The result is inferred by the following information:

- Have profile picture or not : Yes
- User's ID : cjys520
- User's fullname : 
- How many words in the biography : 0
- Have url or not : No
- Private or not : No
- How many posts : 0
- How many followers : 0
- How many followees : 9

自動判斷該帳號是否為騷擾帳號

- 模型效能
  - Accuracy: 0.925
  - Precision: 0.932
  - Recall: 0.917
  - AUROC: 0.99



# 精彩影片片段自動截取

- Generate highlight clips and thumbnails for videos based on bullet-screen (彈幕) information
- Our model is better than the software used by video streaming companies

Table 1: Users' evaluation on the representativeness of the outputted video clips

	All users	Group 1	Group 2
Busk	52.12%	67.38%	47.45%
Stiller	47.88%	32.62%	52.55%

Table 2: Users' evaluation on the representativeness of the outputted thumbnail images

	All users	Group 1	Group 2
Busk	47.62%	63.08%	43.39%
Stiller	52.38%	36.92%	56.61%

- Group 1: users who are familiar with the videos
- Group 2: others

# 影片「片段」搜尋

- Search for the video clips inside long videos based on bullet-screen (彈幕) information

Film Segment Search

## 搜尋結果

JOJO的奇妙冒險 不滅鑑石（僅限台灣地區）：第14話 || 14:00/24:00

活動作品【小姐姐們】蔡徐坤自黑，Lisa燃炸舞台！，1x分鐘帶你看完《青春有你2》高能場面 || 2:00/1:4

JOJO的奇妙冒險 不滅鑑石（僅限台灣地區）：第20話 || 23:00/24:00

活動作品【老邪吐槽】崩潰級RAP！逆天吐槽全员诗朗诵的《青春有你2》 || 4:00/7:12

JOJO的奇妙冒險 不滅鑑石（僅限台灣地區）：第8話 || 10:00/24:00

活動作品【小姐姐們】蔡徐坤自黑，Lisa燃炸舞台！，1x分鐘帶你看完《青春有你2》高能場面 || 1:00/1:4

活動作品【老邪吐槽】崩潰級RAP！逆天吐槽全员诗朗诵的《青春有你2》 || 3:00/7:12

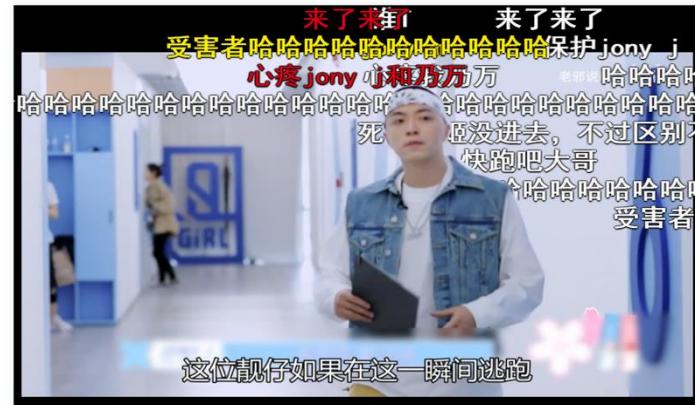
活動作品【小姐姐們】蔡徐坤自黑，Lisa燃炸舞台！，1x分鐘帶你看完《青春有你2》高能場面 || 4:00/1:4

活動作品【小姐姐們】你最受不了淡黃長裙還是冰清玉潔？快來pick《青春有你2》最煩的選手吧！ || 3:00/1:36

活動作品 【老邪吐槽】崩潰級RAP！逆天吐槽全员诗朗诵的《青春有你2》

娛樂 > 綜藝 2020-03-28 11:01:39 最高全站日排行49名

248.4萬播放 · 4.0萬彈幕 未经作者授權，禁止轉載



# 從 FAQ 自動產生 Chatbot

- 從常見問題集 (FAQ) 自動產生客服對話機器人
- 對話機器人利用 Elasticsearch、Word2Vec、及 BERT 判斷「使用者的問題」與「常見問題集 中各問題」的相似度
- 對話機器人回傳相似的常見問題及答案



重設密碼

Top 3 matches:

- (1) 我是畢業生，忘記密碼，無學生證認證身分，該如何修改密碼？
- (2) 我的帳號仍未失效，要如何更改密碼？
- (3) 我是教職員身分，忘記密碼該如何處理？



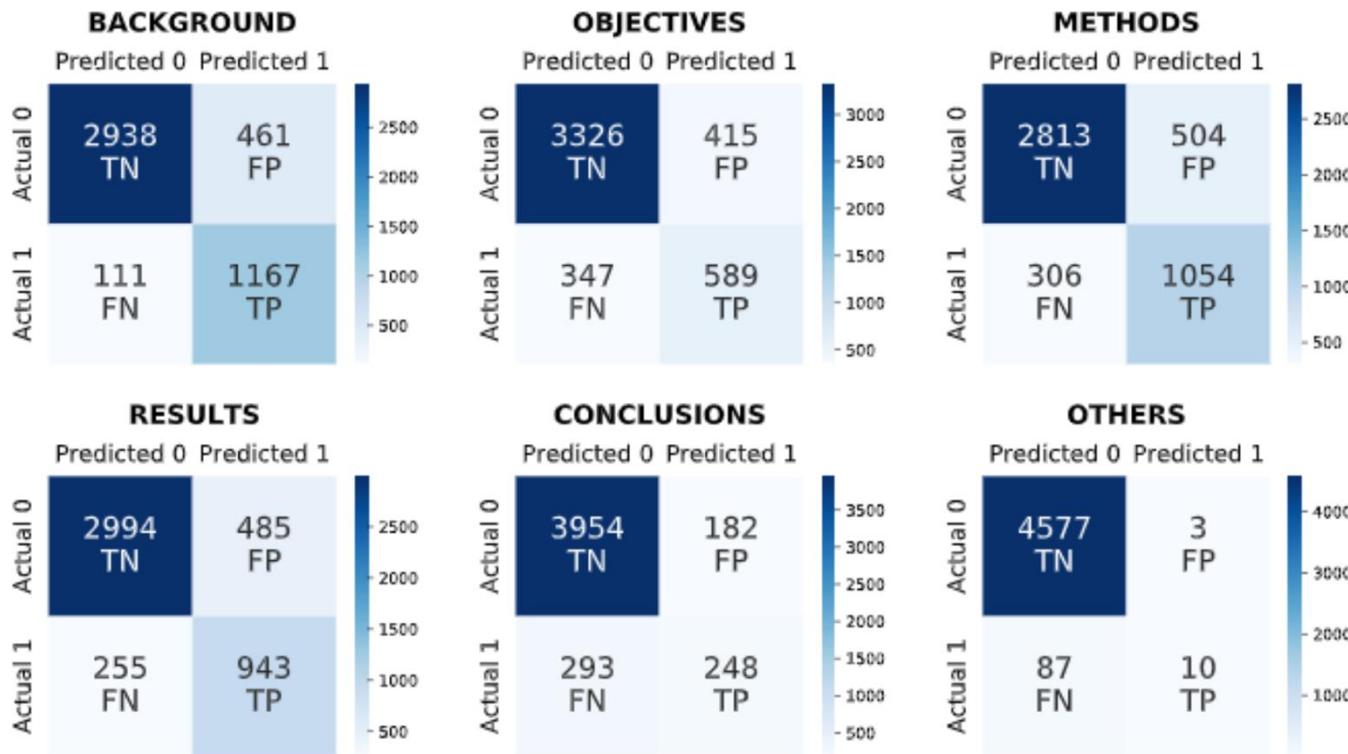
# PDF 數學式解析器

- 要讓電腦“瞭解”文件中的數學式，第一步需要讓電腦能解析數學式
  - E.g.,  $(a + b)^2 \Rightarrow (a + b)^2$
- PDF是科學論文最常見的格式
- 為了在不同裝置能有一樣的文件外觀，PDF 描述每個符號應該以怎樣的型式 (e.g., 大小、字型、顏色等) 出現在哪個位置
  - 這使得數學式很難被自動化的解析
- 我們採機器學習+自訂規則解析PDF中的數學式



# 論文摘要的句子之撰寫目的預測

- Predict the “purposes” of each sentence in an abstract



# 我是大明星 – 明星臉分析

